LLMs4OL 2024: The 1st Large Language Models for Ontology Learning Challenge at the 23rd ISWC
LLMs4OL 2024 Task Participant Papers
https://doi.org/10.52825/ocp.v4i.2486

Published: 02 Oct. 2024

The Ghost at LLMs4OL 2024 Task A: Prompt-Tuning-Based Large Language Models for Term Typing

Thiti Phuttaamart[1], Natthawut Kertkeidkachorn[2], and Areerat Trongratsameethong[1]

[1]Chiang Mai University, Chiang Mai, Thailand

[2]Japan Advanced Institute of Science and Technology, Japan

*Correspondence: Thiti Phuttaamart, thiti_ph@cmu.ac.th

Abstract: The LLMs4OL Challenge @ ISWC 2024 aims to explore the intersection of Large Language Models (LLMs) and Ontology Learning (OL) through three main tasks: *1) Term Typing*, *2) Taxonomy Discovery* and *3) Non-Taxonomic Relation Extraction*. In this paper, we present our system's design for the term typing task. Our approach utilizes automatic prompt generation using soft prompts to enhance term typing accuracy and efficiency. We conducted experiments on several datasets, including WordNet, UMLS, GeoNames, NCI, MEDCIN, and SNOMEDCT_US. Our approach outperformed the baselines on most datasets, except for GeoNames, where it faced challenges due to the complexity and specificity of this domain, resulting in substantially lower scores. Additionally, we report the overall results of our approach in this challenge, which highlight its promise while also indicating areas for further improvement.

Keywords: Large Language Models, Ontology Learning, Prompt Tuning

1 Introduction

Currently, most information on the World Wide Web is in a format that is readable and understandable by humans, but computers require significant processing to comprehend this data. To address this, the Semantic Web has been introduced, extending the capabilities of the World Wide Web to make information on the internet interpretable and interconnected more efficiently. This is achieved using Ontology, which models the concepts of information within a specific domain. Typically, creating an ontology is complex, time-consuming, and requires domain expertise. Therefore, Ontology Learning, which automates the extraction and creation of structured data from unstructured information, has been employed. Given the rapid development of Large Language Models (LLMs) with their deep understanding of language, the LLMs4OL Challenge [1] aims to explore and utilize these models to facilitate automatic ontology creation. The LLMs4OL Challenge comprises three tasks.

1. Term Typing: Discover the generalized type for a lexical term
2. Taxonomy Discovery: Discover the taxonomic hierarchy between type pairs

3. Non-Taxonomic Relation Extraction: Identify non-taxonomic and semantic relations between types

In this study, we are participating in term typing task. The goal of Term Typing task is to assign types to lexical terms. For instance, given the term TUXIS POND from the GeoNames dataset, the correct type would be ”lake”. For the term typing task, previous methods have primarily focused on using prompts with specific templates to identify term types. However, the key challenge lies in finding an effective prompt that produces accurate results. To address this issue, we propose a prompt-tuning-based LLM for term typing, utilizing automatic prompt generation with soft prompts to enhance both the accuracy and efficiency of the task. The repository of our approach is publicly available (https://github.com/themes12/Prompt-Tuning-for-LLMs4OL).

2 Related Work

Ontology learning is a technique used to extract knowledge from unstructured text and create structured data known as an ontology. Popular ontology learning methods include using lexico-syntactic patterns [2] and clustering methods [3], or employing lexico-syntactic patterns for term and relation extraction and clustering methods for type discovery [4]. Additionally, seed-term-based bootstrapping methods are also employed [5]. Recently, LLMs have been utilized in ontology learning and have produced promising results [6]. Nevertheless, this method relies on using specific hard prompts, which are difficult to craft and may not yield optimal results. To address these challenges, soft prompting techniques, such as prompt tuning [7], have been developed. Soft prompts involve creating learnable vectors, often referred to as virtual tokens, that are prepended to the input embeddings and further refined through training. Unlike hard prompts, soft prompts do not require manual crafting, making them more flexible and easier to adapt to different tasks.

3 Approach

We designed the system, which consists of two phases as shown in Figure 1: 1) Training and 2) Testing. In the training phase, we begin with a dataset containing terms and their types. The data are preprocessed to remove any characters that might cause issues, and then combined with an initial prompt. This input is fed into the LLM to create a fine-tuned prompt. During the testing phase, the fine-tuned prompt is used on new, unseen data, where the terms have no specified types. The LLM predicts the most appropriate type for each term, and the results are formatted for evaluation, ensuring

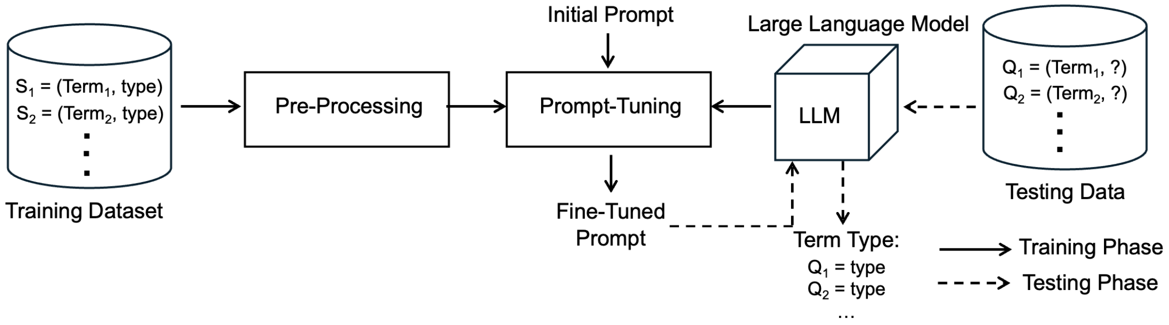

Figure 1. *The design of the system*

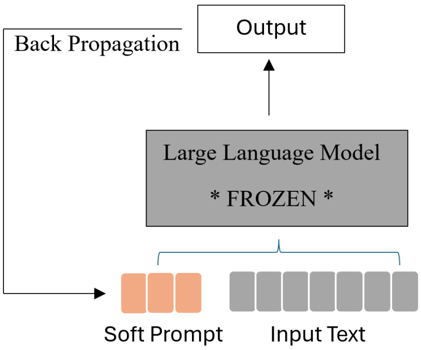

Figure 2. *Prompt tuning.*

accuracy and alignment with the expected output format. The details of each phase are as follows:

3.1 Training Phase

The training phase involves two main steps. The details of each step are as follows:

3.1.1 Pre-processing

The objective of this preprocessing step is to remove characters that could interfere with the final output, particularly during the process of splitting the output by commas to convert it into a list for multi-label classification. This step is essential for ensuring compatibility with the AutoModelForCausalLM and the evaluation system. For instance, in datasets like Gene Ontology, some labels contain commas (e.g., ”regulation of alternative mRNA splicing, via spliceosome”), which could disrupt the output if not handled correctly. By removing problematic characters, we can prevent such issues and maintain the integrity of the results. Additionally, the inputs are restructured into a format suitable for training by tokenizing and padding to ensure uniform input length.

3.1.2 Soft Prompting

The objective of the soft prompting step is to efficiently adapt LLMs to perform specific downstream tasks without the need to retrain the entire model for each task. Training LLMs requires a significant amount of time and resources. One effective way to enable a LLM to perform specific downstream tasks is through the use of prompts. Prompts help to describe the task or provide examples of the task (few-shot). There are two types of prompts.

1. Hard prompt involves manually creating the prompt by hand. The downside is that it requires substantial effort to create a good prompt.
2. Soft prompt involves creating a vector, referred to as virtual tokens, and prepend them to the input embeddings for further training with the dataset. The drawback is that humans cannot read the prompt.

In this study, we employ soft prompt. There are various techniques for creating soft prompts, each designed for different tasks. For example, prefix tuning was designed for natural language generation tasks, while P-tuning is designed for natural language understanding tasks. Multitask prompt tuning is another technique that learns a single prompt from data for multiple task types. We have chosen to use the prompt tuning technique because it was initially developed for text classification tasks on T5 models.

This makes it particularly well-suited for our application, as it leverages the strengths of prompt-based methods in handling classification tasks efficiently. The process begins with an initial prompt, which provides a basic template or instruction set for the task. This initial prompt is then refined and adapted during the training process to become a fine-tuned prompt. The advantage of using prompts is that there is no need to train a separate model for each downstream task. Instead, a single LLM can be utilized, greatly reducing the required time.

3.2 Testing Phase

The testing phase is designed to evaluate the performance of the fine-tuned LLM on new, unseen data. This phase involves feeding the model with testing data and analyzing its output to determine its accuracy and effectiveness in predicting term types. Once the testing data is prepared, it is fed into the fine-tuned prompt and subsequently into the LLM. The model processes the input terms and generates predictions for their types. The fine-tuned prompt guides the model to understand the context and requirements of the task, leveraging the knowledge gained during the training phase.

4 Experiment

4.1 Datasets

The datasets used in the term typing task [8] consist of the following four sub-datasets:

1. **WordNet.** WordNet is a lexicosemantics dataset derived from the original WordNet. It contains 40,943 terms for training and 9,470 terms for testing, encompassing four types: nouns, verbs, adverbs, and adjectives.
2. **GeoNames.** GeoNames includes data on geographical locations, with 8,078,865 instances for training, 702,510 instances for testing, and a total of 680 classes.
3. **UMLS.** The UMLS dataset comprises three sub-datasets:
   - **NCI.** Created by NCI Enterprise Vocabulary Services (EVS) to standardize vocabulary for organizational and public use. It includes terms related to clinical care, translational and basic research, public information, and administrative activities, with 96,177 instances for training and 24,045 for testing, covering 125 classes.
   - **MEDCIN.** Contains medical terminology such as symptoms, medical history, physical examination findings, diagnostic tests, diagnoses, and treatment options, with 277,028 instances for training and 69,258 for testing, spanning 87 classes.
   - **SNOMEDCT_US.** A foundational general terminology used in electronic health records (EHRs), with 277,028 instances for training and 69,258 for testing, encompassing 87 classes.
4. **Gene Ontology.** This dataset includes three sub-ontologies:
   - **Biological Process.** Describes biological processes occurring in living organisms at the cellular level, with 195,775 instances for training and 108,300 for testing, across 792 classes.
   - **Cellular Component.** Describes the positions or structures within a cell, with 228,460 instances for training and 126,485 for testing, covering 323 classes.

Table 1. *MAP@1 Scores for Our Approach Compared to the Baseline Across Datasets*

	WordNet	**GeoNames**	**NCI**	**MEDCIN**	**SNOMEDCT_US**
Baseline	0.9170	**0.4330**	0.3280	0.5180	0.4340
Our Approach	**0.9368**	0.3863	**0.6009**	**0.7397**	**0.6707**

- **Molecular Function.** Describes the activities of gene products, with 196,074 instances for training and 107,432 for testing, spanning 401 classes.

After that, we split the data into 90% for training and 10% for validation in the WordNet, UMLS, and Gene Ontology datasets. For the GeoNames dataset, due to its large size, we split the data into 99% for training and 1% for validation. During the prompt tuning process, the UMLS and Gene Ontology datasets are sampled to 50,000 instances, and the GeoNames dataset is sampled to 100 instances per class. The entire WordNet dataset is used as it is.

4.2 Experimental Setup

Our study investigates a range of LLMs, including BLOOM-1B7, BLOOM-3B, BLOOM-7B1, LLaMA-7B, LLaMA-2-7B-HF, LLaMA-2-7B-CHAT-HF, Meta- Llama-3-8B, Meta-Llama-3-8B-Instruct, BioMistral-7B, and LLaMA-OpenBioLLM-8B. Based on the results from the validation datasets, we selected the following models for each dataset: BLOOM-3B for WordNet, NCI, and SNOMEDCT_US; Meta-Llama-3-8B-Instruct for Geo Names and Biological Process; BLOOM-1B7 for MEDCIN; and BioMistral-7B for Cellular Component and Molecular Function. We implemented the models using Auto-ModelForCasualLM and set the hyperparameters as follows: learning rate: $3e-2$, epochs: 2-4, train size: 15% for WordNet and 5% for GeoNames, and 30% for other datasets. The max token length is 10, and the virtual token size is 15 for WordNet, 40 for GeoNames, 30 for UMLS, 30 for Biological Process and Molecular Function, and 29 for Cellular Component. The choice of models and hyperparameters is based on the results obtained from experiments on the validation datasets [1].

We used the best results presented in the study [6] as the baseline. Please note that only WordNet, UMLS, GeoNames, NCI, MEDCIN, and SNOMEDC_US were investigated. For evaluation metrics, we use MAP@1 (Mean Average Precision at rank 1) [6] to compare our results with the baseline. MAP@1 measures the precision of the top-ranked result for each query, providing an assessment of the model's effectiveness in retrieving the most relevant results. For reporting the results of our approach on this challenge, we use the standard metrics of precision, recall, and F1 score as provided by the challenge organizers.

5 Result and Discussion

Table 1 presents the MAP@1 scores for our approach compared to the baseline, using the same datasets and evaluation metrics as described in LLMs4OL: Large Language Models for Ontology Learning [6]. Our approach shows enhanced performance across datasets such as WordNet, NCI, MEDCIN, and SNOMEDCT_US, indicating improved term retrieval precision. However, the results for GeoNames reveal persistent challenges related to place name ambiguity. The results of the term typing task across different datasets are summarized in Table 2. The results indicate that the system performs well on the WordNet, NCI, SNOMEDCT_US, and MEDCIN datasets. However, in

[1] https://github.com/themes12/Prompt-Tuning-for-LLMs4OL/blob/main/result-validation.pdf

Table 2. *The result on term typing task*

Dataset	F1	Precision	Recall
WordNet	0.9392	0.9389	0.9395
GeoNames	0.1489	0.1461	0.1519
NCI	0.5370	0.4450	0.6769
MEDCIN	0.5328	0.4183	0.7336
SNOMEDCT_US	0.5275	0.4266	0.6910
Cellular Component	0.1877	0.1653	0.2171
Biological Process	0.1025	0.0964	0.1095
Molecular Function	0.1270	0.1278	0.1261

the NCI, SNOMEDCT_US, and MEDCIN datasets, the recall is significantly higher than the precision, which may be due to class imbalance. The performance on the GeoNames and Gene Ontology datasets is significantly worse. For GeoNames, the problem may stem from the ambiguity of place names and the fact that these names are often proper nouns, making them difficult to predict. Additionally, datasets like the Biological Process dataset, which has 792 classes, or the Geonames dataset, with 680 classes, are more challenging compared to smaller datasets like WordNet, which has only 4 classes, or the NCI dataset, with 125 classes. The larger number of classes in these bigger datasets can make predictions harder. For the Gene Ontology dataset, the poor results may be due to the biological nature of the data, which includes information on genes, molecules, and structures. This domain is highly specialized and contains a vast number of possible classes.

6 Conclusion

In this study, we explored the use of soft prompt tuning for the term typing task as part of the LLMs4OL Challenge @ ISWC 2024. Our approach demonstrated strong performance on several datasets, particularly WordNet and UMLS sub-datasets (NCI, MEDCIN, SNOMEDCT_US), indicating the viability of soft prompt tuning for ontology learning tasks. However, the results on GeoNames and Gene Ontology datasets were less satisfactory, highlighting challenges such as class imbalance and the complexity of specialized domains. To improve the results, future work could focus on incorporating additional contextual information beyond just the term, which may help the LLM make better predictions. Additionally, employing techniques other than soft prompts, such as Retrieval-Augmented Generation (RAG), could enhance the LLM's ability to access up-to-date knowledge and external information, potentially leading to improved prediction capabilities. These strategies could address the current limitations and further advance the effectiveness of soft prompt tuning for ontology learning tasks.

Author contributions

Thiti Phuttaamart: Software; Writing – Original Draft Preparation; Conceptualization.
Natthawut Kertkeidkachorn: Conceptualization; Writing - Review Editing; Project administration; Supervision.
Areerat Trongratsameethong: Writing - Review Editing; Project administration; Supervision.

Competing interests

The authors declare that they have no competing interests.

Acknowledgement

This work was supported by JSPS Grant-in-Aid for Early-Career Scientists (Grant Number 24K20834).

References

[1] H. Babaei Giglou, J. D'Souza, and S. Auer, "Llms4ol 2024 overview: The 1st large language models for ontology learning challenge," *Open Conference Proceedings*, vol. 4, Oct. 2024.

[2] M. Hearst, *Automated discovery of wordnet relations." wordnet an electronic lexical database*, 1998.

[3] L. Khan and F. Luo, "Ontology construction for information selection," in *14th IEEE International Conference on Tools with Artificial Intelligence, 2002.(ICTAI 2002). Proceedings.*, IEEE, 2002, pp. 122–127.

[4] J. Watróbski, "Ontology learning methods from text-an extensive knowledge-based approach," *Procedia Computer Science*, vol. 176, pp. 3356–3368, 2020.

[5] C. H. Hwang, "Incompletely and imprecisely speaking: Using dynamic ontologies for representing and retrieving information.," in *KRDB*, Citeseer, vol. 21, 1999, pp. 14–20.

[6] H. Babaei Giglou, J. D'Souza, and S. Auer, "Llms4ol: Large language models for ontology learning," in *The Semantic Web – ISWC 2023*, T. R. Payne, V. Presutti, G. Qi, *et al.*, Eds., Cham: Springer Nature Switzerland, 2023, pp. 408–427, ISBN: 978-3-031-47240-4.

[7] B. Lester, R. Al-Rfou, and N. Constant, "The power of scale for parameter-efficient prompt tuning," *arXiv preprint arXiv:2104.08691*, 2021.

[8] H. Babaei Giglou, J. D'Souza, S. Sadruddin, and S. Auer, "Llms4ol 2024 datasets: Toward ontology learning with large language models," *Open Conference Proceedings*, vol. 4, Oct. 2024.