Large Language Models for Creation, Enrichment and Evaluation of Taxonomic Graphs

Viktor Moskvoretskii [a,b,*], Ekaterina Neminova [b], Alina Lobanova [b], Alexander Panchenko [a,c] and Irina Nikishina [d]
[a] *Skoltech, Russia*
E-mails: v.moskvoretskii@skol.tech, panchenko@airi.net
[b] *HSE University, Russia*
E-mails: v.moskvoretskii@skol.tech, esneminova@edu.hse.ru, alobanova@edu.hse.ru
[c] *AIRI, Russia*
E-mail: panchenko@airi.net
[d] *Universität Hamburg, Germany*
E-mail: irina.nikishina@uni-hamburg.de

Abstract. Taxonomies play a crucial role in organizing knowledge for various natural language processing tasks. Recent advancements in LLMs have opened new avenues for automating taxonomy-related tasks with greater accuracy. In this paper, we explore the potential of contemporary LLMs in learning, evaluating and predicting taxonomic relations across multiple lexical semantic tasks. We propose novel method for taxonomy-based instruction dataset creation, encompassing multiple graph relations. With the use of this datasetwe build TaxoLLaMA, a unified model fine-tuned on datasets exclusively based on English WordNet 3.0, designed to handle a wide range of taxonomy-related tasks such as Taxonomy Construction, Hypernym Discovery, Taxonomy Enrichment, and Lexical Entailment. The experimental results demonstrate that TaxoLLaMA achieves state-of-the-art performance on 11 out of 16 tasks and ranked second on 4 other tasks. We also explore LLM ability for constructed taxonomies graph refinement and present comprehensive ablation study and thorough error analysis supported by both manual and automated techniques.

Keywords: Taxonomy, Taxonomy Enrichment, Taxonomy Construction, Lexical Entailment, WordNet, Lexical Semantic

1. Introduction

The Semantic Web extends the current web by enabling machines to understand and respond to complex human requests based on the meaning (semantics) of the information, rather than just matching keywords. Central to this vision is the structuring of data in a way that allows for meaningful interconnections between different data points. Taxonomies, which classify and organize concepts into hierarchical structures, are essential to this process. They provide the backbone for organizing information in a way that is both

*Corresponding author. E-mail: v.moskvoretskii@skol.tech.

accessible and meaningful. By categorizing data into well-defined classes and relationships, taxonomies facilitate the creation of ontologies, which are more complex frameworks that define the relationships between concepts in the Semantic Web. These ontologies enable more accurate data retrieval, allowing for richer, more nuanced interactions with web content. In essence, taxonomies serve as the building blocks of the Semantic Web, providing the necessary structure for data. The integration of taxonomies into the Semantic Web framework enhances the web's ability to handle complex queries, making it a more powerful tool for knowledge discovery and data management.

More formally, taxonomy is a directed acyclic graph that organizes concepts through various relationships, with each node representing a specific concept connected to others via IS-A relations. A prime example of such a taxonomy is WordNet for the English language [57]. WordNet not only includes nodes but also provides definitions, multiple lemmas, and unique sense numbers to distinguish between different meanings within the same synset.

The use of taxonomies is well-justified in various NLP tasks, including Entity Linking [24], Named Entity Recognition [81], and several others [53, 87].

Despite the widespread adoption of LLMs, taxonomies continue to be constructed and curated primarily through the manual efforts of expert linguists. Earlier neural approaches to natural language processing have struggled to automate this task effectively, but this limitation may not apply to the latest generation of LLMs. While some research has shown that Transformer models underperform in this area, these studies were conducted using much less powerful language models than those available today [36, 67].

Recent analyses of LLMs highlight their impressive capacity to internally store vast amounts of knowledge [51, 77, 80]. Additionally, as these models have scaled, they have demonstrated emerging in-context learning abilities, enabling rapid adaptation to new tasks [28]. These observations suggest that LLMs could be effectively leveraged for lexical semantic tasks. However, despite some previous attempts to apply LLMs in this domain, research remains limited, and the challenges are significant. The few studies that have explored LLMs for lexical semantics have primarily focused on hyponymy and hypernymy relationships, with little attention given to other types of graph relations [21, 63, 65]. Moreover, these studies have generally been limited to hypernym discovery, neglecting the broader range of tasks that taxonomies can support. For instance, research on Taxonomy Enrichment often uses LLMs only to extract representations that are then fed into Graph Neural Networks, rather than directly employing LLMs for the full range of tasks [46].

In this paper, we aim to fill the gap in existing research by exploring how modern foundation models can learn and apply taxonomy graph relations across multiple lexical semantic tasks. Specifically, we focus on using a single LLM to tackle four distinct tasks simultaneously: Taxonomy Construction, Hypernym Discovery, Taxonomy Enrichment, and Lexical Entailment. We hypothesize that contemporary LLMs, when pretrained exclusively on the English WordNet, can effectively learn taxonomy relations by leveraging their inherent language knowledge and align it with the established human-labeled structure.

To sum up, the contribution of the paper is as follows:

- We investigate the ability of LLMs to learn taxonomic structures and predict entities at any level within a taxonomy.
- We introduce a novel dataset creation method that encompasses a variety of taxonomy-related subtasks, including hypernym prediction, hyponym prediction, insertion between two existing nodes, and synset mixing, expanding beyond previous setups that focused solely on hypernym prediction.
- We develop several instructional datasets based exclusively on English WordNet-3.0 for training a taxonomy-based LLM. Additionally, we gather definitions for input words in the Taxonomy Enrichment and Lexical Entailment datasets using sources like Wikidata[1] and ChatGPT[2].
- With the use of aforementioned dataset, we introduce TaxoLLaMA, a unified model tailored to handle a wide range of lexical-semantic tasks, achieving state-of-the-art results in 11 out of 16 tasks and securing second place in 4 tasks.

[1] http://wikidata.org
[2] https://chat.openai.com

- We conduct a comprehensive error analysis across all tasks using both manual and automated methods, including the evaluation of error patterns and model performance with the assistance of ChatGPT.
- We demonstrate the capability of LLMs to refine existing taxonomies by incorporating multiple relationships they have learned.

We also make data, code and models publicly available.[3]

This work is an extended version of the work described in conference papers [59, 60]. The novelty of this particular work compared to the previous versions is as follows:

- We examine the ability of LLMs to resolve graph cycles using learned relations and explore the benefits of this procedure.
- We investigate how LLMs can leverage multiple relations to refine an already constructed graph.
- We explore the advantages of utilizing bidirectional relations to enhance the refinement of constructed taxonomies.
- We extend our Taxonomy Construction results to include the Food subset.

2. Related Work

In this section, we provide a brief overview of previous approaches to the lexical semantics tasks that are the focus of our experiments. We explore the development of graph and taxonomy construction methods and discuss the challenges where taxonomic knowledge has shown to be particularly advantageous.

Taxonomies & LLMs research mostly had focused on encoder-based rather than GPT-style models for taxonomy learning. Notable examples include CTP [19, 25, 36]. Most studies involving LLMs in taxonomy construction have explored the use of models like LM-Scorer [43], which employs BERT [27] and RoBERTa [55] among masked LMs, and GPT-2 [67] among causal LMs. These studies typically employ zero-shot sentence probing or experiment with prompts for taxonomy learning. However, their results have not surpassed the state-of-the-art GNN models for tasks like TexEval-2. Notably, there is a lack of research comparing these methods to more recent open-source models such as LLaMA-2 [82] and Mistral [45] for taxonomy-related tasks, that is the part of the current paper.

Hypernym Discovery task involves generating a list of hypernyms for a given hyponym, as illustrated in Figure 2a. A recent contribution in this area is a taxonomy-adapted, fine-tuned T5 model introduced by [65]. Prior to this, several approaches have been explored. The 300-sparsans method [10] improves upon the traditional word2vec technique. The Hybrid model [37] combines the k-Nearest Neighbor method with Hearst patterns. CRIM [11], recognized as the best performer in the SemEval competition, uses a Multilayer Perceptron (MLP) structure with a contrastive loss function. Lastly, the Recurrent Mapping Model (RMM) [9] employs an MLP with residual connections and a contrastive-like loss function.

Taxonomy Enrichment involves determining the most suitable position for a missing node within a taxonomy, addressed in SemEval-2016 Task 14 [48]. Over the past few years, various architectures have been developed to tackle this task. TMN [92] uses multiple scoring mechanisms to identify ⟨hypernym, hyponym⟩ pairs for a given query concept. TaxoEnrich [46] utilizes two LSTM networks [76] to encode information about both ancestors and descendants. Additionally, TaxoExpan [73] employs a Graph Neural Network (GNN) [71] to predict whether the query concept is a child of an anchor concept.

Taxonomy Construction task is focused on building a domain taxonomy starting from a raw list of terms. Previously, this task was solved with the use of GNN, such as Graph2Taxo [72] or employing zero-shot language model for scoring pairs or mask token probability, such as LMScorer and RestrictMLM [43]. However, some approaches differ with focus on Hearst patterns boosted with Poincare embeddings for refinement.

[3] https://github.com/uhh-lt/lexical_llm

Lexical Entailment involves classifying the semantic connections between word pairs. For instance, if we consider the term “tiger” (a hyponym), it inherently suggests the broader category “big cat” (a hypernym).

Recent research in lexical entailment includes various innovative models. SeVeN [30] encodes relationships between words, while Pair2Vec [47] and a modified GloVe approach from [44] utilize word co-occurrence vectors along with Pointwise Mutual Information to understand semantic connections. The LEAR model [85], on the other hand, fine-tunes Euclidean space to better reflect hyponymy-hypernymy relationships. Graph-based approaches, the “Global” Entailment Graph (GBL) [39] employs a GNN focusing on local learning, while its evolution, the “Contextual” Entailment Graph (CTX) [40], enhances this by integrating contextual link prediction. The CTX model was later improved with an entailment smoothing technique proposed by [56], which currently holds SoTA for this task.

3. Methodology

In this section, we describe the process of building an instruction-tuning dataset specifically designed for taxonomy learning using LLMs and further fine-tuning.

3.1. Dataset Collection Algorithm

The dataset creation process is largely based on the English WordNet 3.0, chosen for its structured and well-maintained organization. Our focus is mainly on the nouns subgraph, not only because it represents the most frequent category in WordNet, but also because recent research [52] has identified it as a challenging class for language models to master.

We begin the process of dataset creation by utilizing a Directed Acyclic Graph (DAG) derived from WordNet, which is structured around "IS A" relationships. Next, we randomly select edges or subsets from this graph, dividing them into different subsets while taking into account all possible tree operations. A comprehensive explanation of the dataset construction algorithm can be found in Subsection 3.1.

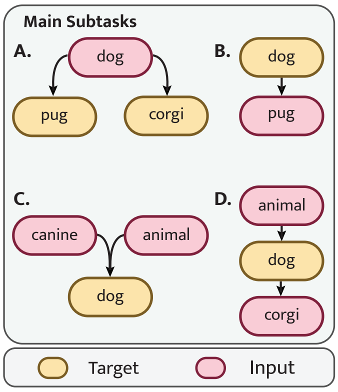
We posit that a diverse dataset encompassing various scenarios offers two key advantages:

- A diverse dataset enhances the model’s ability to generalize, enabling it to understand broader relationships between words across a wide range of subtasks.
- A diverse dataset also empowers the model to develop and apply different strategies for constructing taxonomies effectively.

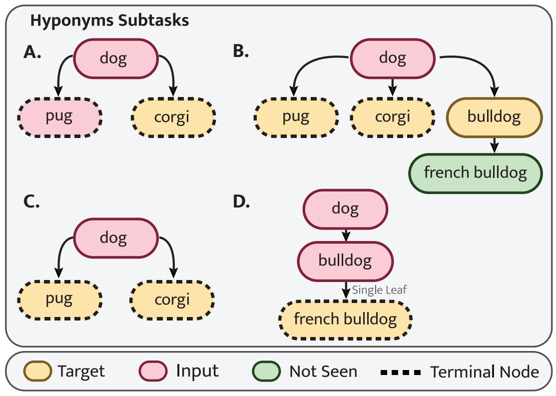
To account for the widest possible range of tree operations within the graph, we gather four distinct subsets, with a particular emphasis on hyponym and hypernym prediction. The tasks include the following scenarios (as illustrated in Figure 1a):

1. **Hyponym prediction** (1a.A): Predicting a list of hyponyms associated with a given synset from the taxonomy.
2. **Hypernym prediction** (1a.B): Identifying the hypernym based on the provided input word.
3. **Synset mixing** (1a.C): Predicting a single hyponym by combining information from two different synsets.
4. **Insertion** (1a.D): Determining a word when given both its hypernym and hyponym.

We ensure that our test and training datasets are completely distinct, with no overlap between them. Specifically, none of the test nodes are included in any of the subtask scenarios. The statistics for each subset are detailed in Table 1.



(a) Examples of IS-A relation structures: (A) hyponym prediction, (B) hypernym prediction, (C) synset mixing, and (D) insertion.



(b) Examples of hyponym subtasks: Leaves Divided (A), Internal Nodes (B), Only Leaves (C), Single Leaves (D).

Fig. 1. Examples for main (a) and hyponym (b) subtasks.

Category	$\text{TaxoLLaMA}_{\text{multi}}$ Train	Test	TaxoLLaMA Train	Test	$\text{TaxoLLaMA}_{\text{bench}}$ Train	Test
Hypernym prediction	1 338	364	44 772	0	36 775	0
Hyponym prediction	16 789	828	0	0	0	0
Synset mixing	1 461	47	0	0	0	0
Insertion	648	35	0	0	0	0
Total	20 236	1 274	44 772	0	36 775	0

Table 1

The statistics of the dataset samples for Taxonomy Learning based on WordNet.

3.1.1. Formal Algorithm

To develop a precise algorithm, we define subtask sets derived from the graph, which are represented as a collection of the following mini-sets:

$$A_i = \{p, \{c_j\}_{j=1}^{deg^+ p}\} \in A,$$

$$B_i = \{p, c\} \in B,$$

$$C_i = \{p_1, p_2, c\} \in C,$$

$$D_i = \{g, p, c\} \in D,$$

Here, c dennotes hyponyms, p - hypernyms, and g - hyperhypernyms.

To facilitate comprehensive set intersections, we introduce the concept of "deep intersection," denoted as $\overline{\cap}$. This operation goes beyond the intersection of individual elements in two sets by considering the intersection between the elements within the subsets of each set. It is mathematically expressed as: $S_1 \overline{\cap} S_2 = \bigcup_{ij}(S_{1i} \cup S_{2j})$.

In the next phase, our goal is to generate random training and testing sets, aiming for approximately 1,000 samples in the test set. We ensure that the training set primarily consists of hyponym and hypernym predictions, while other types of samples are evenly distributed. This task is challenging due to the potential for significant overlap among different cases and the sequence in which samples are collected. To manage this complexity, we introduce a distribution over subtasks, denoted as $\mathbb{P}_{data}$. This allows us to manually

Algorithm 1 Dataset collection algorithm

Input: Sets A, B, C, D sampled from Graph
Output: Train and Test Sets

```
Train := Empty Array
Test := Empty Array
Collect sets A, B, C, D.
while (A ∪ B ∪ C ∪ D) ≠ ∅ do
    cur_set ~ P_data
    cur_sample = cur_set.pop()
    if cur_sample^t ∩̄ Train = ∅ then
        to_test ~ P_test
        if to_test == 1 then
            Test.append(cur_sample)
        else
            Train.append(cur_sample)
        end if
    else
        Train.append(cur_sample)
    end if
end while
```

adjust the probability of sampling each subtask, giving us greater control over the composition of the dataset.

To regulate the likelihood of samples being allocated to the test set, a Bernoulli distribution was considered, denoted as $\mathbb{P}_{test}$, with a parameter p . In the purpose of each relationship is effectively learned, we manually found optimal values for these probabilities:

For $\mathbb{P}_{data}$: $P(A) = 0.51$, $P(B) = 0.39$, $P(C) = 0.05$, and $P(D) = 0.05$.

For $\mathbb{P}_{test}$: $p = 0.05$ and $q = 0.95$.

During data collection, we utilize the "pop()" operation, which removes and returns the last element from a set.

To manage the complexities associated with dominant word categories, we perform a topological sort on the graph. We then ensure that no vertex in our sets has a level lower than a specified parameter, referred to as "level." This condition is expressed as: $\forall i, S \quad \forall v \in S_i : \quad TopSort(v) \geqslant level$. For our collected data, we set level $= 3$.

We also designate a "target" vertex for each element within the subtasks. This enables us to monitor the inclusion of this specific target vertex in the test set, ensuring the integrity of our evaluation process. The definitions of these "target" vertices vary depending on the subtask and can be outlined as follows:

- $A_i^t = \{c_j\}$: The focus is on tracking all hyponyms. If the hyponyms haven't been encountered in the training set, the target cannot be determined. However, encountering the hypernym in the test set is acceptable since it is provided in the prompt.
- $B_i^t = c$: If the hyponym hasn't been seen, it indicates that this pair has not been encountered. Otherwise, the hyponym would have been added to the tracking. Therefore, if the hyponym is unseen, it implies the corresponding edge has not been observed.
- $C_i^t = c$: Similarly, if the hyponym hasn't been seen, it indicates that the target hasn't been observed.
- $D_i^t = p, c$: This scenario involves tracking two edges: $g - p$ and $p - c$, analogous to cases A and B. This restriction ensures that both edges are handled appropriately.

3.2. Downstream Efficient Dataset

Through a thorough testing procedure that considered different probabilities for sampling relations, along with extensive ablation of our choices, we found that the most efficient approach, in terms of downstream performance, was to learn only the hypernymy relation. Formally, this is equivalent to setting $P(B) = 1$ and all other probabilities to zero. Considering the differences we refer to the optimal model as TaxoLLaMA, to the optimal model, that has test data nodes completely excluded from training set as TaxoLLaMA$_{\text{bench}}$ and model encoding multiple relations as TaxoLLaMA$_{\text{multi}}$. We acknowledge that this procedure trains the model to encode only the hypernymy relation. However, it appears sufficient to outperform all other settings. We believe this is likely due to the nature of the learning procedure and the relative simplicity of hypernymy compared to other types of relations.

In addition to the data collected through the described algorithm, we incorporate definitions for child nodes from WordNet to help disambiguate the sense of the input word. Since definitions might not be available for certain subtasks during inference—such as Lexical Entailment, MAG PSY, and MAG CS in the context of Taxonomy Enrichment—we also generate definitions using ChatGPT for test sets that lack pre-existing explanations or source them from Wikidata.

For generating definitions, we used the web interface of ChatGPT 3.5 (February 2024) and the "gpt-3.5-turbo" model from the same period. The prompts used for these requests, along with the statistics of the generated definitions, are detailed in the Appendix A, specifically in Examples 9-10 and Table 13. This step is crucial, as experiments have shown that the absence of definitions can significantly reduce the model's performance [59].

3.3. Model Finetuning

For our research, we employ the most widely used foundational language models, specifically Llama2-7B and Mistral-7B. We chose to exclude smaller models like GPT-2 from our analysis due to their minimal performance across all subsets. To optimize these models, we applied a 4-bit quantization technique. Subsequently, we fine-tuned them using LoRA [41] for one training epoch with a batch size of 64. We used the AdamW optimizer with a learning rate of 3×10^{-4}, coupled with a cosine annealing scheduler. For any additional fine-tuning, the models were trained with a reduced batch size of 2.

Our inputs include an LLaMA-2 system prompt that looks as follows:

(1) [INST] «SYS» You are a helpful assistant. List all the possible words divided with a comma. Your answer should not include anything except the words divided by a comma «/SYS»

Then we introduce a technical-style input prompt and the expected output format:

(2) hypernym: dog.n.1 | hyponyms: [/INST]

(3) pug, corgi,

We also explore the impact of altering the style of the prompt with numerical representations, lemmas, and definitions: *"dog.n.1"*, *"dog (dog, domestic dog, Canis familiaris)"*, *"dog (a member of the genus Canis that has been domesticated by man since prehistoric times)"*.

After experimenting with various types of additional information that could enhance the prompt—such as lemmas, definitions, and synset numbers—we finalized the system prompt that specifies the desired output (4). This prompt is combined with an input word selected from WordNet, along with its definition (5), and the target (6), which represents the true hypernym of the input word, also obtained from WordNet. Below, we present a sample from our dataset used for instruction tuning of TaxoLLaMA. This training sample includes the following components:

(4) [INST] «SYS» You are a helpful assistant. List all the possible words divided with a comma. Your answer should not include anything except the words divided by a comma «/SYS»

(5) **hyponym**: tiger (large feline of forests in most of Asia having a tawny coat with black stripes) | **hypernyms**: [/INST]

(6) big cat,

Model	Hyponym	Hypernym	Insertion	Synset Mixing	Mean
GPT2	0.006	0.033	0.018	0.027	0.021
TaxoLLaMA$_{\text{multi}}$ Numbers	0.099	0.267	0.262	**0.239**	0.162
TaxoLLaMA$_{\text{multi}}$ Lemmas	**0.127**	0.293	0.329	0.218	0.188
TaxoLLaMA$_{\text{multi}}$ Definitions	0.123	0.494	**0.436**	0.234	**0.247**
Mistral-7B Definitions	0.085	**0.498**	0.436	0.160	0.218

Table 2

Fine-tuned models MRR scores on the test set. Bold represents the best result, underlined are second-ranked

4. Instruction Taxonomy Learning Results

In this Section, we describe the results for Taxonomy Learning with TaxoLLaMA$_{\text{multi}}$ which comprises the algorithm for dataset creation, different model templates, fine-tuning, and evaluation, that further allowed us to get the optimal recipe for TaxoLLaMA. We also perform an ablation study to understand how different the performance is for widespread common knowledge words and terms.

We evaluate the performance of our models using the Mean Reciprocal Rank (MRR), a metric that indicates the rank position of the first correct answer. We chose MRR over other ranking metrics because they might impose overly stringent criteria. To assess the models, we create a list of potential candidates, each separated by a comma, and then match these candidates with the target words.

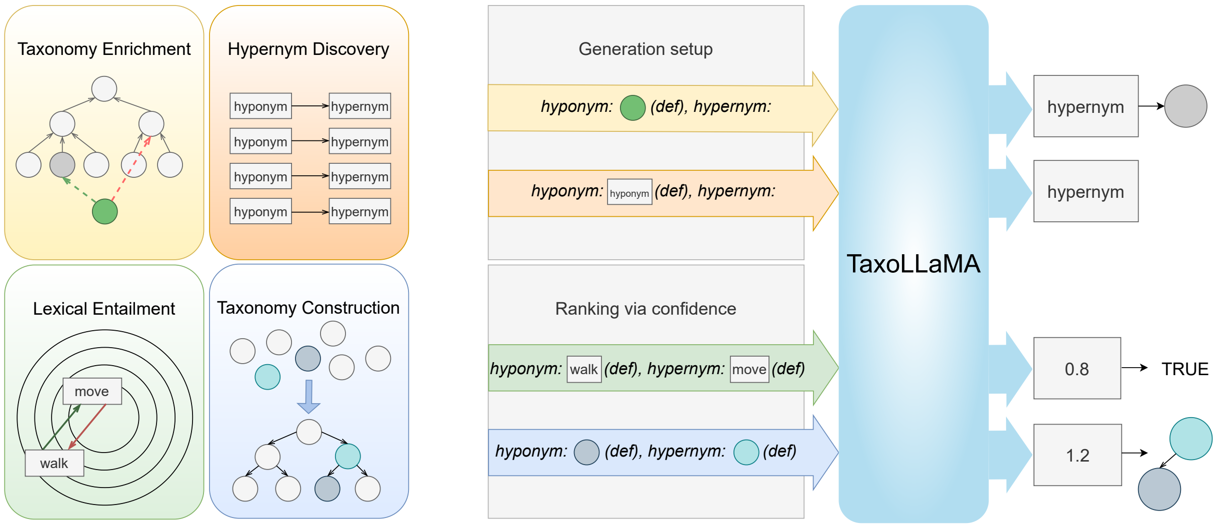
In our preliminary experiments, we conducted a case study to evaluate ChatGPT's performance on the task. Despite using few-shot learning techniques, ChatGPT consistently failed to provide correct answers. For example, when prompted with the word "Maltese," the model suggested "dog breed" and "animal" as hypernyms, missing the correct WordNet hypernym, "toy dog." Similarly, for the term "machine translation," it generated hypernyms like "automated translation" and "language translation system," whereas the correct hypernyms according to WordNet are "artificial intelligence" and "computational linguistics." These results show that while ChatGPT could identify the general domain, it struggled to pinpoint the exact synset from WordNet. Notably, this was an area where our fine-tuned model excelled, highlighting its superior performance in these specific instances.

The results of our fine-tuned models are summarized in Table 2. The best performance is observed with hypernym prediction and insertion tasks, in which on average, the correct answer is the second candidate suggested by the model. However, a closer look through manual error analysis reveals that this score is an average of cases where the model correctly identifies the first candidate and cases where it fails to provide any correct candidates at all. The result for other tasks are lower nearly twice or more, suggesting that those tasks are much more complex for LLM.

Incorporating lemmas significantly improves results compared to using numbers, with the highest scores achieved when definitions are included. This improvement may be attributed to the autoregressive nature of the model's generation process. By providing a more relevant context, the model's output distribution shifts closer to the correct answers. Including definitions appears to either strengthen this shift or make it more precise.

We believe that the size of the model is the main contributing factor, rather than pre-training data amount. Despite using lemmas or definitions for disambiguation, the score does not change drastically for worst cases, showing that disambiguation is not the key problem. Moreover, the underperformance may be linked to the sequential nature of LM loss in instruction tuning. With multiple correct answers, it poses a problem to properly apply loss, as different orders of correct nodes would imply completely different loss values. The problem usually arises with hyponym prediction.

Considering different models, GPT2 fails to learn any task and Mistral-7B and LLaMA2-7B are nearly equal with slightly better average LLaMA performance, however Mistral has been reported to outperform LLaMA-2 in other benchmarks [45].



(a) Lexical semantic tasks

(b) Generation and ranking pipelines for solution of various lexical semantic tasks

Fig. 2. Examples with input and output for each task are highlighted by color. Rectangle "hypernym" denotes a word generated by the model; circle means a node from the graph. Confidence score determines the existence of a relationship between the two nodes provided in the input.

5. Downstream Tasks Application

In this section, we aim to extend the evaluation of the LLM's ability to learn taxonomic relations and explore the capabilities of all TaxoLLaMA versions in addressing four tasks that require taxonomic knowledge: Taxonomy Construction, Hypernym Discovery, Taxonomy Enrichment, and Lexical Entailment.

We hypothesize that a model trained with taxonomic knowledge will be effective in solving taxonomy-related tasks. To test this hypothesis, we apply the previously mentioned family of TaxoLLaMA models, to these tasks using two adaptation strategies:

Ranking approach is applied to Taxonomy Construction and Lexical Entailment. This technique involves assessing the hypernymy relation through perplexity calculations, where a lower perplexity score indicates a stronger relationship. We also apply some additional evaluation with LLM for each task, that is described in corresponding sections.

Generative approach directly employs the procedure used during training. Starting with a hyponym, the model generates a list of potential hypernyms. This approach is utilized for the Hypernym Discovery and Taxonomy Enrichment datasets.

5.1. Taxonomy Construction

We test the TaxoLLaMA versions on the downstream task: SemEval 2016 Task 13. We use the Eurovoc taxonomies ("Science", "Environment") and Wordnet "Food" from SemEval-2016 [12]. These datasets are commonly used as a benchmark for testing models' abilities of taxonomy construction.

To create the taxonomy, we use a ranking approach for every possible edge and leave only those below the optimal threshold. We have not used definitions, as they are not given. Extended experiments on hyponymy template construction, definitions simulation and other construction techniques are thoroughly described in Section 6.3. We also apply self-refinement based on hypernymy perplexity to resolve self-loops and delete multiple parental edges. The refinement procedure is described and ablated in Section 6.2.

	TexEval-2 best	TAXI+	Graph2Taxo pure	Graph2Taxo best	LMScorer	RestrictMLM	TaxoLLaMA	TaxoLLaMA$_{\text{bench}}$	TaxoLLaMA$_{\text{multi}}$
Science	31.3	41.4	39.0	**47.0**	31.8	37.9	44.55	42.36	44.12
Environment	30.0	30.9	37.0	40.0	26.4	23.0	**45.13**	**44.82**	**42.03**
Food	36.01	34.1	-	-	24.9	24.9	**51.71**	**51.18**	42.35

Table 3

F1 score for the Taxonomy Construction Task for Science, Environment and Food domain dataset

	1A: English	2A: Medical	2B: Music	1B: Italian	1C: Spanish
CRIM* [11]	36.10	54.64	60.93	-	-
Hybrid* [37]	34.07	64.47	77.24	-	-
RMM* [9]	39.07	54.89	74.75	-	-
T5 [65]	45.22	44.73	53.35	24.04	27.50
300-sparsans* [10]	-	-	-	25.14	37.56
TaxoLLaMA zero-shot	38.05	43.09	42.7	1.95	2.21
TaxoLLaMA$_{\text{bench}}$ zero-shot	37.66	42.2	44.36	1.47	2.08
TaxoLLaMA fine-tuned	**54.39**	**77.32**	**80.6**	**51.58**	**57.44**
TaxoLLaMA$_{\text{bench}}$ fine-tuned	**51.59**	**73.82**	**78.63**	**50.95**	**58.61**

Table 4

MRR performance on Hypernym Discovery. * refers to the systems that rely on the provided dataset only, without LLM pretraining or additional data being used. *Zero-shot* is trained on the WordNet data only, without fine-tuning on the target dataset.

5.2. Results and Discussion

In Table 3, we showcase the F1-scores for the Science, Environment and Food datasets. We evaluate our three models version against earlier methods.

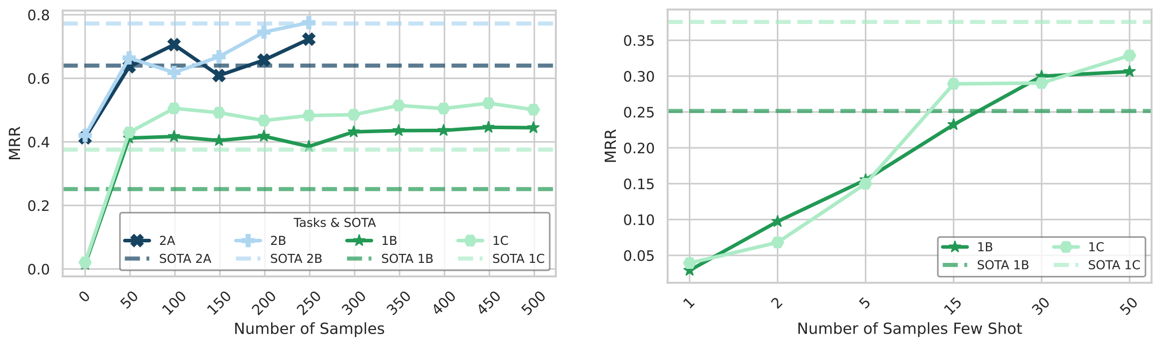
Our results indicate that our method outperforms all existing models on the Environment and Food domains and ranks second on the Science domain. The top-performing approach for the "Science" dataset, Graph2Taxo [72], achieves its best score through a GNN-based cross-domain transfer framework, specifically during their ablation study. Interestingly, the framework's default setup does not produce the highest scores (refer to [72] (pure) in Table 3). In the current study, we do not focus on specific building strategies, but rather on refinement, reflected in Section 6.2, therefore TaxoLLaMA could be researched more on that aspect. It is also clear that zero-shot LM performed the worst on average, underscoring the need for specific fine-tuning and stronger models [43].

5.3. Hypernym Discovery

We evaluate TaxoLLaMA on the Hypernym Discovery task from SemEval-2018 [14] using a generative approach. This task includes an English test set for general hypernyms, as well as two domain-specific sets for “Music” and “Medical.” Additionally, there are general test sets available for Italian and Spanish. The performance is assessed using the Mean Reciprocal Rank (MRR) metric. We employ a zero-shot approach, where the model is tested without fine-tuning on the training datasets. Notably, the test set is distinct from WordNet and may require multiple hops to reach hypernyms, making it suitable for both general and narrow domains.

5.3.1. Results

The results for the English language, presented in Table 4, show that both the fine-tuned TaxoLLaMA and TaxoLLaMA$_{\text{bench}}$ models significantly surpass previous SoTA results. Although the zero-shot performance of our models is somewhat lower than their fine-tuned counterparts, they still achieve outcomes comparable to earlier results in general English tasks and remain competitive in domain-specific tasks, despite the fact that previous methods were all fine-tuned.



(a) Fine-tuning

(b) Few-shot learning

Fig. 3. Experiments for domain and language adaptation on the Hypernym Discovery datasets.

Multilingual Performance In the case of Italian and Spanish, the fine-tuned model exceeds previous SoTA results. This success might be attributed to the model's inherent multilingual capabilities, given that LLaMA-2 was initially designed to be multilingual, even though fine-tuning was conducted solely on English pairs. However, the zero-shot performance reveals challenges in generating accurate hypernyms for languages other than English. It is important to note that Italian and Spanish data were not part of the instruction tuning dataset.

Zero-shot Performance To better understand the underperformance in zero-shot scenarios, we analyzed the impact of fine-tuning across different domains and languages, as depicted in Figure 3a. The analysis shows that, apart from task 2B, the model surpasses previous SoTA results with as few as 50 samples for fine-tuning. Furthermore, the varying scores emphasize the model's sensitivity to the quality and characteristics of the training data.

Few-shot Performance We further investigated the few-shot learning approach for Italian and Spanish to evaluate the model's adaptability in an in-context learning setting, as depicted in Figure 3b. The model surpassed previous SoTA benchmarks for Italian, showing a near-logarithmic improvement with 30 and 50 shots, but did not perform as well for Spanish. We attribute this suboptimal few-shot performance to the 4-bit quantization and the relatively small model size. Smaller models generally underperform on various benchmarks compared to their larger counterparts, as demonstrated by the example of LLaMA-2 [83]. Moreover, smaller or quantized models have limited capacity compared to larger models, a finding supported by earlier research [29, 33, 54, 88]. As it has been already seen [54], the benefits of few-shot learning are less pronounced in quantized models compared to full-precision models.

5.4. Taxonomy Enrichment

Following the methodology of previous studies [46, 92], the task is considered as ranking graph nodes based on their probability of being the correct hypernym. The aim is to position the correct hypernyms at the top of the ranking, ensuring the node is accurately placed within the taxonomy. In our approach, we utilize the generative method, as shown in Figure 2b.

The Taxonomy Enrichment benchmark includes datasets such as WordNet Noun, WordNet Verb, MAG-PSY, and MAG-CS [46, 73]. To maintain consistency with the TaxoExpan test set [73], we selected 1,000 nodes from each dataset. In line with [46], we utilize scaled MRR [90] as the key evaluation metric. This metric is derived by multiplying MRR by 10 and then averaging it across all correct hypernyms associated with each node.

To improve disambiguation, we created definitions for MAG datasets that lacked predefined explanations, either by generating them with ChatGPT or retrieving them from Wikidata. We utilized the ChatGPT 3.5 web interface and the "gpt-3.5-turbo" model, both from February 2024, for generating these definitions.

	MAG-CS	MAG-PSY	Noun	Verb
TaxoExpan [73]	19.3	44.1	39.0	32.5
GenTaxo [91]	23.9	46.4	28.6	42.8
TMN [92]	24.3	53.1	36.7	35.4
TaxoEnrich [46]	**57.8**	**58.3**	44.2	45.2
TaxoLLaMA zero-shot	7.4	7.3	n/a	n/a
TaxoLLaMA$_{\text{bench}}$ zero-shot	8.5	6.6	n/a	n/a
TaxoLLaMA fine-tuned	24.9	29.8	**48.0**	**52.4**
TaxoLLaMA$_{\text{bench}}$ fine-tuned	30.2	31.4	**45.9**	**51.9**

Table 5

Scaled MRR Across Tasks for Taxonomy Enrichment. Here, "n/a" stands for "not applicable", as TaxoLLaMA has already seen WordNet data and its performance cannot be considered as zero-shot. *Zero-shot* is trained on the WordNet data only, without fine-tuning on the target dataset.

The prompts used and the statistics related to the generated definitions are provided in Appendix A, specifically in Examples 9-10 and Table 13. This step is essential, as missing definitions can lead to a decrease in model performance, as highlighted in [59].

5.4.1. Results

The results in Table 5 indicate that our model outperforms all previous approaches on the WordNet Noun and WordNet Verb datasets. However, it falls short of the current SoTA method on the more specialized MAG-CS and MAG-PSY taxonomies, even with fine-tuning. Interestingly, TaxoLLaMA$_{\text{bench}}$, despite having access to less data, unexpectedly delivered better performance on the MAG datasets. To gain further insight into the reasons for the overall underperformance, we conducted an in-depth error analysis, which is discussed in Section 7.1.

5.5. Lexical Entrailment

For our evaluation, we rely on the Hyperlex benchmark [86] alongside the ANT entailment subset [34], which serves as a detailed refinement of the Levy/Holt dataset [38].

ANT Dataset features sentence pairs that differ by a single argument within their syntactic structure (e.g., "The audience *applauded* the comedian" versus "The audience *observed* the comedian," as shown in Table 2 of [34]). Each pair is classified into one of several relationships: antonymy, synonymy, directional entailment, or non-directional entailment (the reverse of directional entailment). For sentences that exhibit an entailment relationship, we treat the differing elements as hypernym-hyponym pairs.

The ranking method here is enriched with confidence scores. The confidence score is the ratio between forward and reversed perplexity. The forward perplexity is the regular one, and the reversed is obtained by first reversing hypernym and hyponym roles.

Based on this confidence scores entailment relations are asssed as the ratio of the hypernym to hyponym ranking scores, with normalization by the L2 norm to estimate the probability of entailment. For example, we compute the perplexity score of "move" as a hypernym of "walk" ($PPL_{m \to w}$) and the reverse ($PPL_{w \to m}$). The ratio $\frac{PPL_{m \to w}}{PPL_{w \to m}}$ between these scores then reflects the model's confidence in the entailment relationship.

Additionally, we developed TaxoLLaMA$_{\text{verb}}$ specifically for this subtask. This model was pre-trained exclusively on verbs from WordNet, with the aim of better capturing the taxonomy structure of verbs.

HyperLex Dataset is designed to assess entailment for both verbs and nouns, using a scale from 0 to 10. A score of 0 signifies no entailment, whereas a score of 10 represents strong entailment. The objective is

	$\mathrm{AUC_N}$	AP
GBL [39]	3.79	58.36
CTX [40]	15.44	65.66
GBL-$\mathrm{P_{K=4}}$ [56]	13.91	64.71
CTX-$\mathrm{P_{K=4}}$ [56]	**25.86**	67.47
TaxoLLaMA zero-shot	0.89	51.61
$\mathrm{TaxoLLaMA_{bench}}$ zero-shot	2.82	54.24
$\mathrm{TaxoLLaMA_{verb}}$ zero-shot	19.28	**69.51**

(a) Performance on the Lexical Entailment ANT dataset. *Zero-shot* is trained on the WordNet data only, without fine-tuning on the target dataset.

Setting	Model	Lexical	Random
fine-tuned	RoBERTa best [66]	79.4	82.8
	RoBERTa mean [66]	65.8	63.8
	LEAR [85]	54.4	69.2
zero-shot	Relative [15]	54.3	58.4
	Pair2Vec [47]	33.4	54.3
	GRV SI [44]	48.3	55.4
	SeVeN [30]	46.9	**62.7**
	FastText	43.9	54.3
	TaxoLLaMA	**70.2**	59.3

(b) Spearman Correlation for lexical and random test subsets of Hyperlex benchmark. *Zero-shot* is trained on the WordNet data only, without fine-tuning on the target dataset.

to maximize correlation with the gold-standard scores. For this dataset, we apply the ranking approach directly, without any additional processing and usage of confidence scores.

Earlier approaches typically generate embeddings and then train a basic SVM on the Hyperlex training set. Fine-tuned models, such as RoBERTa, require significant computational resources and are specifically adapted to the Hyperlex dataset. In contrast, our zero-shot model utilizes perplexities directly as predictions, eliminating the need for any additional training. As a result, direct comparisons may not fully account for the distinct methodologies and resource demands, highlighting the importance of evaluating each method within its own specific context.

5.5.1. Results

Results on the ANT Dataset The results presented in Table 6a compare our models with previous SoTA performances on the ANT dataset. A significant observation is the clear disparity in performance between TaxoLLaMA, trained on both nouns and verbs, and $\mathrm{TaxoLLaMA_{verb}}$, specialized exclusively in verbs.

$\mathrm{TaxoLLaMA_{verb}}$ outperforms TaxoLLaMA in the Lexical Entailment task, indicating potential challenges in processing nouns and verbs together, which may hinder effective verb learning. This could be related to the constraints of quantization and LORA adapter tuning. Interestingly, this issue appears to be specific to the entailment task, as it does not arise in other tasks like Taxonomy Enrichment, which also involves a verb dataset. The discrepancy might be due to the metrics used, which require precise normalized perplexity rankings.

Table 6a reveals that $\mathrm{TaxoLLaMA_{verb}}$ attains SoTA performance in Average Precision and ranks second in normalized AUC. However, it is important to note that the comparison with previous SoTA results is somewhat imbalanced, as the top-performing models leveraged additional Entailment Smoothing [56] to enhance their performance. This technique has not yet been applied to our models, suggesting a potential avenue for future improvements.

Results on the HyperLex Dataset Table 6b highlights the effectiveness of our model, outperforming the previous SoTA in a zero-shot scenario for the “Lexical” subset and securing second place for the “Random” subset. Interestingly, while most models tend to perform better on the random subset, our approach deviates from this trend, indicating that the larger training size of the random subset may provide greater advantages to other methods. Despite the simplicity of our zero-shot method, it still delivers impressive results. Future research could investigate incorporating this score as a meta-feature in task-specific models, or refining our entire model for better alignment.

6. Ablation Study

In this section, we explore the ability of $\text{TaxoLLaMA}_{\text{multi}}$ to build taxonomy graph and further refine it with several strategies and inspect poor performance for learning hyponymy relation with $\text{TaxoLLaMA}_{\text{multi}}$ with thorough ablation study.

	Hyponym	Internal Nodes	Leaves Divided	Only Leaves	Single Leaves	Insertion	Hypernym	Synset Mixing	Mean
Numbers	-0.036	0.001	-0.046	-0.240	0.006	0.079	-0.055	0.089	-0.005
Lemmas	-0.036	-0.019	-0.004	-0.124	-0.016	-0.028	-0.053	0.120	-0.001
Definition	-0.042	-0.044	0.029	-0.194	0.000	0.025	0.006	0.097	0.054

Table 7

MRR Scores difference between easy and hard subsamples (easy−hard) for the taxonomy learning subtasks. Green color denotes that scores are higher for the "easy" subset, Red color shows that better results are for the "hard" subset.

6.1. Subtypes of Hyponyms

To provide better understanding of undergoing processes, we splitted the hyponymy cases into more detailed. The narrower case, reflected at Figure 1b are as follows:

– **Leaves Divided** (1bA): all hyponyms required to be terminal nodes, with half of them passed as input, other half is considered a target.
– **Internal Nodes** (1bB): Hyponyms are required to have at least one internal node.
– **Only Leaves** (1bC): all target hyponyms are terminal nodes.
– **Single Leaves** (1bD): hyponyms are required to be terminal nodes, and they are the only hyponyms for the node.

The results in Table 8 show that terminal nodes are predicted better as internal. We believe that this results stems from ambiguity of internal nodes, as we noted through manual examination of them. The main issue with predicting internal nodes (1bB), is prediction of more distant nodes (with hop $\geqslant 2$) instead of the direct hyponyms. Additionally, the scenario in (1bA) shows lower performance compared to predicting all possible hyponyms (1bC). This suggests that the key issue is not simply ambiguity, as it would be resolved with cohyponyms, but rather the model's difficulty in generating the appropriate hyponyms. The model's predictive scope seems constrained by the candidates provided in the input. The scenario involving a single leaf hyponym (1bD) proves to be particularly challenging to predict, even when hypernyms are provided as input. This difficulty might be due to the complexity and relative rarity of such instances in natural language, making them harder for the model to learn and generate accurately.

6.1.1. Common Words VS Terminology

To better understand the consistently low average results, we closely examined the model outputs and found that the complexity of the dataset could be a significant factor. Some synsets within the WordNet taxonomy may be overly specialized, which poses a challenge for the model when predicting hyponyms or hypernyms. To investigate this possibility, we categorized our dataset into two distinct groups: commonly known words (classified as the *"easy"* category) and more specialized terms, jargon, or rare words (classified as the *"hard"* category). This categorization was carried out with the help of three computational linguistics experts, who annotated the test set. They were asked to classify a sample as *"hard"* if it contained at least one word that could be considered a term, jargon, or rare, and as *"easy"* if it did not. The level of agreement among the annotators, as measured by Krippendorff's alpha, was 0.67, which is high enough to consider the annotations valid and reliable.

We revisit the performance metrics for both the "easy" and "hard" subsets and summarized the results in Table 7. Interestingly, models generally performed better on the "hard" nodes, especially when it came to predicting hyponyms. However, when using our best model that incorporates word definitions, the "easy"

Model	2A	2B	2C	2D
#Samples	*117*	*115*	*110*	*486*
Numbers	0.152	0.113	0.220	0.068
Lemmas	**0.179**	0.154	0.220	**0.100**
Definition	0.175	**0.163**	**0.268**	0.081

Table 8

MRR scores for the LlaMA-2 model with a different hyponyms prediction subtasks, with column names correspond to Figure 1b.

Subset	Graph Type	2 Parents	3 Parents	4 Parents	5 Parents	6 Parents	7 Parents	8 Parents	9 Parents	10+ Parents
Environment	Noisy Graph	6	10	13	11	8	13	12	17	141
	Optimal Graph	35	18	2	1	-	-	-	-	-
Science	Noisy Graph	6	5	6	6	13	11	10	10	38
	Optimal Graph	5	-	-	-	-	-	-	-	-

Table 9

Distribution of Parent Counts across Graph Types and Subsets

instances yielded higher scores, particularly in cases that did not involve hyponym predictions. This trend, though, is not consistent across all prompt types; in some cases, "hard" instances were more accurately predicted, even when dealing with hypernyms or internal nodes.

We believe the results of the ablation study suggest that the model tends to predict less common words more accurately. This could be because the candidate pool for these terms is smaller, allowing the model to focus more directly on the correct answers. Additionally, the model likely encounters these rare words less frequently and typically within consistent, specific contexts, which might enhance its predictive accuracy for such terms.

6.2. Self-refinement for constructed graph

In this section, we explore how the pre-trained TaxoLLaMA$_{\text{multi}}$ can refine the constructed graph using learned taxonomy graph relations. We address several issues in the existing graph and present solutions along with the corresponding results.

6.2.1. Multiple Parental Nodes

Typically, having multiple parent nodes in taxonomies and ontologies is rare, usually with no more than three parents. We analyzed how our LLM constructs the graph across various thresholds, with the results presented in Table 9. The findings show that assigning multiple parents is common when using non-optimal thresholds, and while less frequent, it still occurs with optimal thresholds.

We addressed the issue of multiple parent nodes using several techniques:

- **Delete All**: Remove all parental edges from nodes that have multiple parents. This approach operates under the assumption that if the LLM is uncertain about a parental relationship, it's better to omit such connections altogether.
- **Perplexity**: Retain only the edge with the lowest perplexity value. This method ensures that only the most probable parental connection, as determined by the LLM, is maintained.
- **Synset Mixing**: Keep two parental nodes if the LLM's synset mixing perplexity falls below a specified threshold. This technique leverages the LLM's ability to combine synsets and assesses the likelihood of such combinations resulting in the target node. We explore two variations:
  - **Synset Mixing One**: Applies the same threshold used during edge construction.
  - **Synset Mixing Two**: Uses a different, specifically chosen threshold for evaluation.

Task	Baseline	Delete All	Perplexity	Synset Mixing Two	Synset Mixing One	Compose	Perplexity (Cycles)	Hyponymy
Environment	0.400	0.404	**0.420**	0.333	0.411	**0.420**	0.41	0.403
Science	0.421	0.434	**0.441**	0.419	0.428	0.434	0.425	0.423

Table 10

F1 Scores for Different Methods for Self-Refinement of LLM

– **Compose**: Combines the Perplexity and Synset Mixing methods. If two parental nodes do not meet the Synset Mixing threshold criteria, the Perplexity method is applied as a fallback.

The results in Table 10 indicate that most of the self-refinement methods for handling multiple parents improve the quality of the graph. However, the simple perplexity rule proves to be the most effective. We believe this is due to the LLM's stronger ability to encode hypernym relations, while its synset mixing capability is less developed, likely due to limited data during pretraining.

6.2.2. Hypernym-hyponym Validation

In this section, we explore how an LLM can validate edges for both hypernymy and hyponymy relations. After constructing the graph using hypernymy, we investigate the impact of removing edges that fall above the hyponymy threshold on the overall quality. The results presented in Figure 4 demonstrate that using hyponymy for graph refinement can be beneficial, though it requires careful calibration. However, it is not as effective as the refinement techniques used for resolving multiple parental nodes.

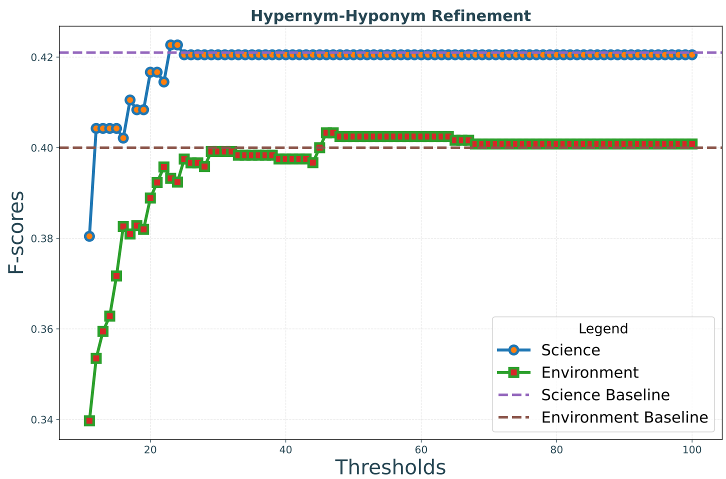


Fig. 4. The graph for hypernym-hyponym validation for Science and Environment. X Axis shows the threshold for hyponymy, Y axis shows the resulting score. Dashed lines indicate initial score without hyponymy validation.

6.2.3. Cycles Resolution

Cycles are typically rare in taxonomies, and self-loops should not exist at all, as they contradict the fundamental structure of taxonomies. To address self-loops and larger cycles, we primarily use the perplexity rule, similar to the approach described in Section 6.2.1, by removing the edge with the highest perplexity.

We also considered eliminating cycles involving three or more nodes by leveraging the LLM's ability to evaluate the insertion of a node followed by the deletion of the least probable connection. However, cycles with three or more nodes are rare when using optimal thresholds and are not included in our analysis, as they consistently result in lower scores compared to the optimal threshold.

The results in Table 10 show an overall improvement with this procedure, particularly in the scientific domain, where closely related concepts are more likely to form loops.

Approach	Method	Template	Sci	Env
$\text{TaxoLLaMA}_{\text{multi}}$ with lemma	brute-force	hyper	<u>0.419</u>	<u>0.409</u>
		hypo	0.192	0.115
	dfs	hyper	0.340	0.213
		hypo	0.137	0.142
$\text{TaxoLLaMA}_{\text{multi}}$ with empty lemma	brute-force	hyper	**0.426**	0.380
		hypo	0.188	0.116
	dfs	hyper	0.338	0.213
		hypo	0.127	0.129
$\text{TaxoLLaMA}_{\text{multi}}$ with numbers	brute-force	hyper	<u>0.416</u>	**0.411**
		hypo	0.185	0.116
	dfs	hyper	0.186	0.186
		hypo	0.125	0.138

Table 11

Results for the downstream TexEval-2 task comparing different fine-tuned models, methods for graph construction, and templates for model inputs. Hyper approach stands for hypernym prediction and hypo for hyponym prediction

6.3. Taxonomy Construction Strategies

Hypernymy vs Hyponymy experiment in Table 11 show that predicting hypernyms performs significantly better than predicting hyponyms, which is coherent with the scores for the respective subtasks during the fine-tuning step.

Construction Methods exploration includes two techniques of building a taxonomy graph. For both of them we traverse through predefined grid and finds the best threshold in terms of metric, however search space is different: the *brute-force* considers all possible edges for the threshold and the *DFS-style* approach starts from root and systematically appends vertices with the threshold.

Results in Table 11 show that brute-force outperformed the DFS-style approach. That could happen due to error accumulation during graph traversal. Incorrect decision on the first couple levels significantly limits our possible edge space.

Prompt was ablated with adding lemmas, empty lemma or specific WordNet number with corresponding models. For prompting with lemmas (as we have no additional lemmas unlike in WordNet), we tried two approaches (duplicate lemma in listing; provide no lemma at all):

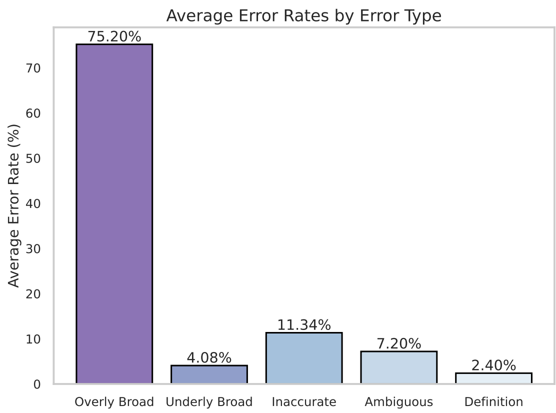
(7) “hypernym: cat (cat) | hyponyms:”

(8) “hypernym: cat () | hyponyms:”

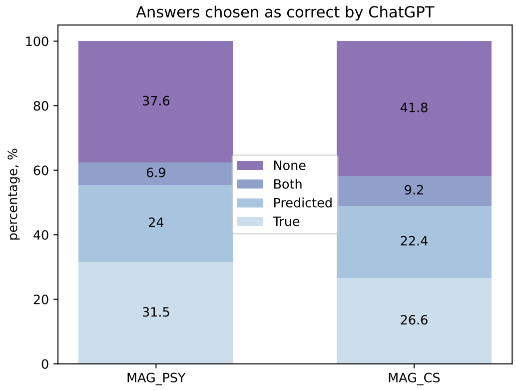
Results in Table 11 show that the best result is obtained with either empty lemma or technical numbers. We believe that model could be distracted when the lemma is repeated, therefore scores are lower. It is unexpected that model with WordNet number has shown outperformance for Environment and Strong result for Science, possibly due to more straightforward task.

7. Error Analysis

In this section, we examine the errors produced by the TaxoLLaMA model, delve into the underlying causes of these inaccuracies, and propose strategies for improving the performance of LLMs when applied to taxonomies.



(a) Average percentage of error types



(b) Automatic evaluation of the MAG datasets using the ChatGPT model.

Fig. 5. (a) Average percentage of error types across Hypernym Discovery and Taxonomy Enrichment datasets. (b) Automatic evaluation of the MAG datasets using the ChatGPT model. The label *"True"* represents the number of instances where ChatGPT favored the gold-standard answers over those generated by TaxoLLaMA; *"Predicted"* indicates cases where ChatGPT preferred the output from TaxoLLaMA. Additionally, ChatGPT could select *"Both"* if it found both answers equally acceptable or *"None"* if neither answer was preferred.

7.1. Hypernym Discovery and Taxonomy Enrichment

Since we use the same generative approach for both Hypernym Discovery and Taxonomy Enrichment, we conduct a combined error analysis. This process is divided into four steps: (i) conducting a manual review to pinpoint the most frequent errors; (ii) performing an automatic error analysis using ChatGPT; (iii) comparing and consolidating the common errors identified; and (iv) classifying these errors with the help of ChatGPT.

We begin by selecting approximately 200 random samples from both the Hypernym Discovery and Taxonomy Enrichment datasets and provide explanations for the model's failure to generate the correct hypernym. Through this process, we identify four categories of errors: (i) predicted hypernyms are excessively broad; (ii) Incorrect or irrelevant definition; (iii) the model fails to produce relevant candidates within the same semantic domain; (iv) miscellaneous cases that do not fit into the other categories.

We use the prompt shown in Example 11 to request that ChatGPT generate potential error types. The resulting output is presented in Example 12, and Table 14 summarizes the error types identified across multiple runs. Afterward, we combine the error types identified both manually and automatically into the following categories:

1. **Overly Broad Predictions**: The model frequently generates predictions that are broader than the intended hypernym.
2. **Overly Narrow Predictions**: Some predictions are too specific and do not capture the generality of the true hypernym.
3. **Inaccurate Predictions**: The model sometimes predicts terms that are semantically similar to the correct hypernym but fails to match the exact wording required.
4. **Conceptual Ambiguity**: The model struggles with input words or concepts that have ambiguous meanings, resulting in incorrect predictions.
5. **Incorrect Definitions**: Errors occur when the model is misled by inaccurate or incorrect definitions retrieved from external sources.

To classify incorrectly predicted instances, we used the prompt provided in Appendix A, as shown in Example 13. The outcomes for each task and dataset are detailed in Table 15 and Figure 5a in Appendix B,

Metric	Science	Environment	Food
Original			
# Nodes	125	261	1486
# Edges	124	261	1533
Constructed			
# Nodes	78	216	1132
# Edges	71	507	1372
# Nodes Missing	48	45	354
# Weak Components	8	5	51
# Nodes w/o original hypernym	4	5	39
# Nodes w/o path to original hypernym	29	70	308
# Nodes w/ path to original hypernym	44	140	784
Mean Distance to original hypernym	1.02	1.15	1.06

Table 12

Statistics of original graph and the constructed graph with highest F1 score. The lower part of the table corresponds to constructed graph statistics

which illustrate the average error distribution. Additionally, Table 16 includes an example corresponding to each type of error. The most prevalent problem, affecting 75% of the cases, is the prediction of overly broad concepts. This issue is likely due to the model's adaptation to domain-specific datasets that are more expansive than WordNet, such as those in the "Music" and "Medical" domains.

In the case of Italian and Spanish, substantial inaccuracies were primarily due to the grammatical complexities inherent in these languages, compounded by dataset limitations, linguistic nuances, and insufficient pre-training data. Likewise, the MAG datasets encountered challenges related to specificity and ambiguity, which resulted in TaxoLLaMA underperforming compared to WordNet-based datasets, as highlighted in Table 5.

A manual review of the MAG taxonomies reveals misclassifications, such as "olfactory toxicity in fish" being incorrectly categorized as a hyponym of "neuroscience." To further evaluate the accuracy of the predicted hypernyms, we leveraged ChatGPT, drawing inspiration from recent research [68]. We provided ChatGPT with the input queries, predicted nodes, and ground truth nodes, asking for a preference. As shown in Figure 5b, ChatGPT often preferred neither of the options, with ground truth hypernyms being favored only slightly more often than the predicted ones. An example of the input query used is detailed in Appendix A, Example 14.

Our evaluation of the overlap between the MAG datasets and WordNet data reveals that they have little in common. Specifically, only 5% of the nodes in the MAG graph are also found in the WordNet graph. The overlap is even less in terms of edges, with only 2% in the CS domain and 4% in the PSY domain matching WordNet connections. Additionally, 92% of the identified connections lack any corresponding path within the WordNet structure. Among the connections that do overlap, we discovered that 28% in CS and 10% in PSY mistakenly identify nodes as their own hypernyms. These disparities highlight why TaxoLLaMA performs less effectively on MAG datasets, as they differ significantly from the WordNet-based data used during training.

In our final analysis, we visualized the embeddings, which highlighted a clear divergence between the predicted outcomes and the actual ground truth within the MAG subsets—a divergence that was not observed in the WordNet data. Detailed findings from this visualization are discussed in Appendix C.

7.2. Taxonomy Construction

Our detailed assessment of the predicted graphs across different domain datasets, based on the data in Table 12, reveals consistent trends. In most cases, the gold standard graphs exhibit a higher number of edges, except for the environment domain. Interestingly, the model tends to miss entire clusters of nodes rather than isolated ones: around 30% of the nodes in the TaxoLLaMA graph are disconnected from their true parents, indicating they belong to separate components.

Although some paths generated by the model are highly accurate, its overall performance is inconsistent—either perfectly on target or completely off course. Frequently, paths with high perplexity are mistakenly discarded, suggesting the model struggles particularly with concepts that are neither highly specific nor overly broad but fall somewhere in the middle of the taxonomy.

This issue is exacerbated by the use of perplexity as a relative metric, where some edges are excluded because they exceed the defined perplexity threshold. However, adjusting the threshold to be more lenient can lead to the creation of incorrect edges. This challenge highlights the need to explore alternative methods, such as employing LLMs as embedding tools, to improve the model's performance.

7.3. Lexical Entailment

Our review of the ANT dataset revealed that it comprises nearly 3,000 test samples but only 589 distinct verbs. This suggests that errors associated with a single verb could potentially be repeated multiple times throughout the dataset. However, when we looked at the overlap with WordNet, we found that only 7 of these verb forms matched.

After lemmatization, the number of unique verbs increases to 338, yet around 42% still cannot be found in WordNet. Moreover, for the verbs that do exist in WordNet, no corresponding paths were identified, which may have negatively impacted the model's performance in this task.

Hyperlex offers more favorable statistics, with nearly 50% of the words being unique and 88% included in WordNet. However, only 27% of the word pairs are represented in the taxonomy, and 99% of these pairs are missing a connecting path.

Perplexity-related errors tend to have high values when dealing with polysemous pairs, such as "spade is a type of card," and low values for synonyms or paraphrases, which indicates semantic closeness without implying a hypernymy relationship. This suggests that the model struggles with lexical diversity and ambiguity, highlighting the necessity of robust disambiguation capabilities in entailment tasks. Additional details are provided in Appendix D.

8. Conclusion

In this paper, we explored the use of LLMs for learning taxonomic relations, evaluating their effectiveness, and applying them to various downstream tasks. To facilitate taxonomy learning, we developed a dataset collection method using WordNet 3.0, which proved highly effective. Our fine-tuned models achieved state-of-the-art performance across several Lexical Semantic tasks, including Taxonomy Construction, Hypernym Discovery, Taxonomy Enrichment, and Lexical Entailment. Specifically, our models secured the top performance in 11 out of 16 tasks and ranked second in 4 others, demonstrating that LLMs are well-suited for solving taxonomy-related challenges.

Additionally, we conducted an extensive ablation study on our model, focusing on the learning of hyponymy by categorizing it into subtypes and levels of difficulty. Our findings revealed that hyponymy is generally more challenging to learn than hypernymy, particularly for concepts located in the middle of the graph. Furthermore, our results suggest that some taxonomy relations are easier to learn for specialized terminology rather than for common concepts. The study also highlighted the potential of LLMs to refine existing taxonomies by utilizing multiple learned taxonomic relations to assess the accuracy of edges, which significantly improved overall performance. In terms of taxonomy construction, our experiments showed that hypernymy plays a crucial role, and that basic, straightforward brute-force methods currently yield the best results.

Lastly, we carried out an in-depth analysis of model errors, revealing inconsistencies between WordNet and other taxonomies, and underscoring the need to revisit and possibly revise MAG taxonomies due to numerous misaligned relations.

References

[1] A.V. Aho and J.D. Ullman, *The Theory of Parsing, Translation and Compiling*, Vol. 1, Prentice-Hall, Englewood Cliffs, NJ, 1972.

[2] R. Aly, S. Acharya, A. Ossa, A. Köhn, C. Biemann and A. Panchenko, Every child should have parents: a taxonomy refinement algorithm based on hyperbolic term embeddings, 2019.

[3] American Psychological Association, *Publications Manual*, American Psychological Association, Washington, DC, 1983.

[4] R.K. Ando and T. Zhang, A Framework for Learning Predictive Structures from Multiple Tasks and Unlabeled Data, *Journal of Machine Learning Research* **6** (2005), 1817–1853.

[5] G. Andrew and J. Gao, Scalable training of L1-regularized log-linear models, in: *Proceedings of the 24th International Conference on Machine Learning*, 2007, pp. 33–40.

[6] N. Arefyev, M. Fedoseev, A. Kabanov and V. Zizov, Word2vec not dead: predicting hypernyms of co-hyponyms is better than reading definitions, in: *Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialogue 2020"*, Moscow, 2020. doi:10.28995/2075-7182-2020-19-13-32. https://www.dialog-21.ru/media/5192/arefyevnvplusetal-152.pdf.

[7] S. Ashkboos, I. Markov, E. Frantar, T. Zhong, X. Wang, J. Ren, T. Hoefler and D. Alistarh, QUIK: Towards End-to-End 4-Bit Inference on Generative Large Language Models, 2023.

[8] H. Bai, T. Wang, A. Sordoni and P. Shi, Better Language Model with Hypernym Class Prediction, in: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 1352–1362. doi:10.18653/v1/2022.acl-long.96. https://aclanthology.org/2022.acl-long.96.

[9] Y. Bai, R. Zhang, F. Kong, J. Chen and Y. Mao, Hypernym Discovery via a Recurrent Mapping Model, in: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, C. Zong, F. Xia, W. Li and R. Navigli, eds, Association for Computational Linguistics, Online, 2021, pp. 2912–2921. doi:10.18653/v1/2021.findings-acl.257. https://aclanthology.org/2021.findings-acl.257.

[10] G. Berend, M. Makrai and P. Földiák, 300-sparsans at SemEval-2018 Task 9: Hypernymy as interaction of sparse attributes, in: *Proceedings of the 12th International Workshop on Semantic Evaluation*, M. Apidianaki, S.M. Mohammad, J. May, E. Shutova, S. Bethard and M. Carpuat, eds, Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 928–934. doi:10.18653/v1/S18-1152. https://aclanthology.org/S18-1152.

[11] G. Bernier-Colborne and C. Barrière, CRIM at SemEval-2018 Task 9: A Hybrid Approach to Hypernym Discovery, in: *Proceedings of the 12th International Workshop on Semantic Evaluation*, M. Apidianaki, S.M. Mohammad, J. May, E. Shutova, S. Bethard and M. Carpuat, eds, Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 725–731. doi:10.18653/v1/S18-1116. https://aclanthology.org/S18-1116.

[12] G. Bordea, E. Lefever and P. Buitelaar, SemEval-2016 Task 13: Taxonomy Extraction Evaluation (TExEval-2), in: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, Association for Computational Linguistics, San Diego, California, 2016, pp. 1081–1091. doi:10.18653/v1/S16-1168. https://www.aclweb.org/anthology/S16-1168.

[13] G. Bordea, P. Buitelaar, S. Faralli and R. Navigli, SemEval-2015 Task 17: Taxonomy Extraction Evaluation (TExEval), in: *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, Association for Computational Linguistics, Denver, Colorado, 2015, pp. 902–910. doi:10.18653/v1/S15-2151. https://www.aclweb.org/anthology/S15-2151.

[14] J. Camacho-Collados, C. Delli Bovi, L. Espinosa-Anke, S. Oramas, T. Pasini, E. Santus, V. Shwartz, R. Navigli and H. Saggion, SemEval-2018 Task 9: Hypernym Discovery, in: *Proceedings of The 12th International Workshop on Semantic Evaluation*, Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 712–724. doi:10.18653/v1/S18-1115. https://www.aclweb.org/anthology/S18-1115.

[15] J. Camacho-Collados, L. Espinosa-Anke, S. Jameel and S. Schockaert, A Latent Variable Model for Learning Distributional Relation Vectors, in: *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, International Joint Conferences on Artificial Intelligence Organization, 2019, pp. 4911–4917. doi:10.24963/ijcai.2019/682.

[16] A.K. Chandra, D.C. Kozen and L.J. Stockmeyer, Alternation, *Journal of the Association for Computing Machinery* **28**(1) (1981), 114–133. doi:10.1145/322234.322243.

[17] J. Chee, Y. Cai, V. Kuleshov and C.D. Sa, QuIP: 2-Bit Quantization of Large Language Models With Guarantees, 2024.

[18] B. Chen, F. Yi and D. Varró, Prompting or Fine-tuning? A Comparative Study of Large Language Models for Taxonomy Construction, *CoRR* **abs/2309.01715** (2023). doi:10.48550/arXiv.2309.01715.

[19] C. Chen, K. Lin and D. Klein, Constructing Taxonomies from Pretrained Language Models, in: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Association for Computational Linguistics, Online, 2021, pp. 4687–4700. doi:10.18653/v1/2021.naacl-main.373. https://aclanthology.org/2021.naacl-main.373.

[20] J. Chen and D. Yang, Unlearn What You Want to Forget: Efficient Unlearning for LLMs, in: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino and K. Bali, eds, Association for Computational Linguistics, 2023, pp. 12041–12052. https://aclanthology.org/2023.emnlp-main.738.

[21] P. Chernomorchenko, A. Panchenko and I. Nikishina, Leveraging Taxonomic Information from Large Language Models for Hyponymy Prediction, in: *Analysis of Images, Social Networks and Texts — 11th International Conference, AIST 2023*, LNCS, Vol. 14486, Springer, 2024. https://link.springer.com/chapter/10.1007/978-3-031-54534-4_4.

[22] Y. Cho, J.D. Rodriguez, Y. Gao and K. Erk, Leveraging WordNet Paths for Neural Hypernym Prediction, in: *Proceedings of the 28th International Conference on Computational Linguistics*, International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 3007–3018. doi:10.18653/v1/2020.coling-main.268. https://aclanthology.org/2020.coling-main.268.

[23] T. Computer, RedPajama: an Open Dataset for Training Large Language Models, 2023. https://github.com/togethercomputer/RedPajama-Data.

[24] L.D. Corro, A. Abujabal, R. Gemulla and G. Weikum, FINET: Context-Aware Fine-Grained Named Entity Typing, in: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, L. Màrquez, C. Callison-Burch, J. Su, D. Pighin and Y. Marton, eds, The Association for Computational Linguistics, 2015, pp. 868–878. doi:10.18653/v1/d15-1103.

[25] A. Davies, J. Jiang and C. Zhai, Competence-Based Analysis of Language Models, *CoRR* **abs/2303.00333** (2023). doi:10.48550/arXiv.2303.00333.

[26] T. Dettmers, A. Pagnoni, A. Holtzman and L. Zettlemoyer, QLoRA: Efficient Finetuning of Quantized LLMs, in: *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. https://openreview.net/forum?id=OUIFPHEgJU.

[27] J. Devlin, M. Chang, K. Lee and K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, in: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, J. Burstein, C. Doran and T. Solorio, eds, Association for Computational Linguistics, 2019, pp. 4171–4186. doi:10.18653/v1/n19-1423.

[28] Q. Dong, L. Li, D. Dai, C. Zheng, Z. Wu, B. Chang, X. Sun, J. Xu, L. Li and Z. Sui, A Survey on In-context Learning, 2023.

[29] V. Egiazarian, A. Panferov, D. Kuznedelev, E. Frantar, A. Babenko and D. Alistarh, Extreme Compression of Large Language Models via Additive Quantization, 2024.

[30] L. Espinosa-Anke and S. Schockaert, SeVeN: Augmenting Word Embeddings with Unsupervised Relation Vectors, in: *Proceedings of the 27th International Conference on Computational Linguistics*, E.M. Bender, L. Derczynski and P. Isabelle, eds, Association for Computational Linguistics, Santa Fe, New Mexico, USA, 2018, pp. 2653–2665. https://aclanthology.org/C18-1225.

[31] L. Espinosa-Anke, F. Ronzano and H. Saggion, TALN at SemEval-2016 Task 14: Semantic Taxonomy Enrichment Via Sense-Based Embeddings, in: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, Association for Computational Linguistics, San Diego, California, 2016, pp. 1332–1336. doi:10.18653/v1/S16-1208. https://www.aclweb.org/anthology/S16-1208.

[32] C. Fellbaum, WordNet, in: *Theory and applications of ontology: computer applications*, Springer, 2010, pp. 231–243.

[33] E. Frantar, S. Ashkboos, T. Hoefler and D. Alistarh, GPTQ: Accurate Post-Training Quantization for Generative Pre-trained Transformers, 2023.

[34] L. Guillou and S.B. de Vroe, ANT dataset, 2023. https://github.com/lguillou/ant.

[35] D. Gusfield, *Algorithms on Strings, Trees and Sequences*, Cambridge University Press, Cambridge, UK, 1997.

[36] M. Hanna and D. Mareček, Analyzing BERT's Knowledge of Hypernymy via Prompting, in: *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, Association for Computational Linguistics, Punta Cana, Dominican Republic, 2021, pp. 275–282. doi:10.18653/v1/2021.blackboxnlp-1.20. https://aclanthology.org/2021.blackboxnlp-1.20.

[37] W. Held and N. Habash, The Effectiveness of Simple Hybrid Systems for Hypernym Discovery, in: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, A. Korhonen, D. Traum and L. Màrquez, eds, Association for Computational Linguistics, Florence, Italy, 2019, pp. 3362–3367. doi:10.18653/v1/P19-1327. https://aclanthology.org/P19-1327.

[38] X. Holt, Probabilistic Models of Relational Implication, 2019.

[39] M.J. Hosseini, N. Chambers, S. Reddy, X.R. Holt, S.B. Cohen, M. Johnson and M. Steedman, Learning Typed Entailment Graphs with Global Soft Constraints, *Transactions of the Association for Computational Linguistics* **6** (2018), 703–717. doi:10.1162/tacl_a_00250. https://aclanthology.org/Q18-1048.

[40] M.J. Hosseini, S.B. Cohen, M. Johnson and M. Steedman, Open-Domain Contextual Link Prediction and its Complementarity with Entailment Graphs, in: *Findings of the Association for Computational Linguistics: EMNLP 2021*, M.-F. Moens, X. Huang, L. Specia and S.W.-t. Yih, eds, Association for Computational Linguistics, Punta Cana, Dominican Republic, 2021, pp. 2790–2802. doi:10.18653/v1/2021.findings-emnlp.238. https://aclanthology.org/2021.findings-emnlp.238.

[41] E.J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang and W. Chen, LoRA: Low-Rank Adaptation of Large Language Models, in: *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*, OpenReview.net, 2022. https://openreview.net/forum?id=nZeVKeeFYf9.

[42] E.J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang and W. Chen, LoRA: Low-Rank Adaptation of Large Language Models, in: *International Conference on Learning Representations*, 2022. https://openreview.net/forum?id=nZeVKeeFYf9.

[43] D. Jain and L. Espinosa Anke, Distilling Hypernymy Relations from Language Models: On the Effectiveness of Zero-Shot Taxonomy Induction, in: *Proceedings of the 11th Joint Conference on Lexical and Computational Semantics*, Association for Computational Linguistics, Seattle, Washington, 2022, pp. 151–156. doi:10.18653/v1/2022.starsem-1.13. https://aclanthology.org/2022.starsem-1.13.

[44] S. Jameel, Z. Bouraoui and S. Schockaert, Unsupervised Learning of Distributional Relation Vectors, in: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, I. Gurevych and Y. Miyao, eds, Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 23–33. doi:10.18653/v1/P18-1003. https://aclanthology.org/P18-1003.

[45] A.Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D.S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L.R. Lavaud, M.-A. Lachaux, P. Stock, T.L. Scao, T. Lavril, T. Wang, T. Lacroix and W.E. Sayed, Mistral 7B, 2023.

[46] M. Jiang, X. Song, J. Zhang and J. Han, TaxoEnrich: Self-Supervised Taxonomy Completion via Structure-Semantic Representations, in: *Proceedings of the ACM Web Conference 2022*, WWW '22, Association for Computing Machinery, New York, NY, USA, 2022, pp. 925–934–. doi:10.1145/3485447.3511935.

[47] M. Joshi, E. Choi, O. Levy, D. Weld and L. Zettlemoyer, Pair2vec: Compositional Word-Pair Embeddings for Cross-Sentence Inference, in: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 3597–3608. https://www.aclweb.org/anthology/N19-1362.

[48] D. Jurgens and M.T. Pilehvar, SemEval-2016 Task 14: Semantic Taxonomy Enrichment, in: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, S. Bethard, M. Carpuat, D. Cer, D. Jurgens, P. Nakov and T. Zesch, eds, Association for Computational Linguistics, San Diego, California, 2016, pp. 1092–1102. doi:10.18653/v1/S16-1169. https://aclanthology.org/S16-1169.

[49] D. Jurgens and M.T. Pilehvar, SemEval-2016 Task 14: Semantic Taxonomy Enrichment, in: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, Association for Computational Linguistics, San Diego, California, 2016, pp. 1092–1102. doi:10.18653/v1/S16-1169. https://www.aclweb.org/anthology/S16-1169.

[50] E. Kasneci, K. Sessler, S. Küchemann, M. Bannert, D. Dementieva, F. Fischer, U. Gasser, G. Groh, S. Günnemann, E. Hüllermeier, S. Krusche, G. Kutyniok, T. Michaeli, C. Nerdel, J. Pfeffer, O. Poquet, M. Sailer, A. Schmidt, T. Seidel, M. Stadler, J. Weller, J. Kuhn and G. Kasneci, ChatGPT for good? On opportunities and challenges of large language models for education, *Learning and Individual Differences* **103** (2023), 102274. doi:https://doi.org/10.1016/j.lindif.2023.102274. https://www.sciencedirect.com/science/article/pii/S1041608023000195.

[51] C. Kauf, A.A. Ivanova, G. Rambelli, E. Chersoni, J.S. She, Z. Chowdhury, E. Fedorenko and A. Lenci, Event knowledge in large language models: the gap between the impossible and the unlikely, *Cognitive Science* **47**(11) (2023), e13386. https://doi.org/10.1111/cogs.13386.

[52] A. Lazaridou, A. Kuncoro, E. Gribovskaya, D. Agrawal, A. Liska, T. Terzi, M. Gimenez, C. de Masson d'Autume, S. Ruder, D. Yogatama, K. Cao, T. Kociský, S. Young and P. Blunsom, Pitfalls of Static Language Modelling, *CoRR* **abs/2102.01951** (2021). https://arxiv.org/abs/2102.01951.

[53] M. Lenz and R. Bergmann, Case-Based Adaptation of Argument Graphs with WordNet and Large Language Models, in: *Case-Based Reasoning Research and Development*, S. Massie and S. Chakraborti, eds, Springer Nature Switzerland, Cham, 2023, pp. 263–278. ISBN 978-3-031-40177-0.

[54] J. Lin, J. Tang, H. Tang, S. Yang, W.-M. Chen, W.-C. Wang, G. Xiao, X. Dang, C. Gan and S. Han, AWQ: Activation-aware Weight Quantization for LLM Compression and Acceleration, in: *MLSys*, 2024. https://doi.org/10.48550/arXiv.2106.09685.

[55] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer and V. Stoyanov, RoBERTa: A Robustly Optimized BERT Pretraining Approach, *CoRR* **abs/1907.11692** (2019). http://arxiv.org/abs/1907.11692.

[56] N. McKenna, T. Li, M. Johnson and M. Steedman, Smoothing Entailment Graphs with Language Models, in: *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, J.C. Park, Y. Arase, B. Hu, W. Lu, D. Wijaya, A. Purwarianti and A.A. Krisnadhi, eds, Association for Computational Linguistics, Nusa Dua, Bali, 2023, pp. 551–563. https://aclanthology.org/2023.ijcnlp-main.37.

[57] G.A. Miller, *WordNet: An electronic lexical database*, MIT press, 1998.

[58] V. Moskvoretskii, F. Anton and K. Denis, IMAD: IMage-Augmented multi-modal Dialogue, 2023.

[59] V. Moskvoretskii, A. Panchenko and I. Nikishina, Are Large Language Models Good at Lexical Semantics? A Case of Taxonomy Learning, in: *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language

Resources and Evaluation (LREC-COLING 2024), N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti and N. Xue, eds, ELRA and ICCL, Torino, Italia, 2024, pp. 1498–1510. https://aclanthology.org/2024.lrec-main.133.

[60] V. Moskvoretskii, E. Neminova, A. Lobanova, A. Panchenko and I. Nikishina, TaxoLLaMA: WordNet-based Model for Solving Multiple Lexical Semantic Tasks, in: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, L.-W. Ku, A. Martins and V. Srikumar, eds, Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 2331–2350. https://aclanthology.org/2024.acl-long.127.

[61] R. Navigli and S.P. Ponzetto, BabelNet: Building a Very Large Multilingual Semantic Network, in: *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Uppsala, Sweden, 2010, pp. 216–225. https://aclanthology.org/P10-1023.

[62] I. Nikishina, V. Logacheva, A. Panchenko and N. Loukachevitch, Studying Taxonomy Enrichment on Diachronic WordNet Versions, in: *Proceedings of the 28th International Conference on Computational Linguistics*, International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 3095–3106. doi:10.18653/v1/2020.coling-main.276. https://aclanthology.org/2020.coling-main.276.

[63] I. Nikishina, A. Vakhitova, E. Tutubalina and A. Panchenko, Cross-Modal Contextualized Hidden State Projection Method for Expanding of Taxonomic Graphs, in: *Proceedings of TextGraphs-16: Graph-based Methods for Natural Language Processing*, D. Ustalov, Y. Gao, A. Panchenko, M. Valentino, M. Thayaparan, T.H. Nguyen, G. Penn, A. Ramesh and A. Jana, eds, Association for Computational Linguistics, Gyeongju, Republic of Korea, 2022, pp. 11–24. https://aclanthology.org/2022.textgraphs-1.2.

[64] I. Nikishina, M. Tikhomirov, V. Logacheva, Y. Nazarov, A. Panchenko and N.V. Loukachevitch, Taxonomy enrichment with text and graph vector representations, *Semantic Web* **13**(3) (2022), 441–475. doi:10.3233/SW-212955.

[65] I. Nikishina, P. Chernomorchenko, A. Demidova, A. Panchenko and C. Biemann, Predicting Terms in IS-A Relations with Pre-trained Transformers, in: *Findings of the Association for Computational Linguistics: IJCNLP-AACL 2023 (Findings)*, J.C. Park, Y. Arase, B. Hu, W. Lu, D. Wijaya, A. Purwarianti and A.A. Krisnadhi, eds, Association for Computational Linguistics, Nusa Dua, Bali, 2023, pp. 134–148. doi:10.18653/v1/2023.findings-ijcnlp.12. https://aclanthology.org/2023.findings-ijcnlp.12.

[66] L. Pitarch, J. Bernad, L. Dranca, C. Bobed Lisbona and J. Gracia, No clues good clues: out of context Lexical Relation Classification, in: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber and N. Okazaki, eds, Association for Computational Linguistics, Toronto, Canada, 2023, pp. 5607–5625. doi:10.18653/v1/2023.acl-long.308. https://aclanthology.org/2023.acl-long.308.

[67] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei and I. Sutskever, Language Models are Unsupervised Multitask Learners (2019).

[68] R. Rafailov, A. Sharma, E. Mitchell, C.D. Manning, S. Ermon and C. Finn, Direct Preference Optimization: Your Language Model is Secretly a Reward Model, in: *NeurIPS*, 2023. https://arxiv.org/abs/2305.18290.

[69] M.S. Rasooli and J.R. Tetreault, Yara Parser: A Fast and Accurate Dependency Parser, *Computing Research Repository* **arXiv:1503.06733** (2015), version 2. http://arxiv.org/abs/1503.06733.

[70] N. Reimers and I. Gurevych, Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, in: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, K. Inui, J. Jiang, V. Ng and X. Wan, eds, Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3982–3992. doi:10.18653/v1/D19-1410. https://aclanthology.org/D19-1410.

[71] F. Scarselli, M. Gori, A.C. Tsoi, M. Hagenbuchner and G. Monfardini, The graph neural network model, *IEEE transactions on neural networks* **20**(1) (2008), 61–80. https://doi.org/10.1109/TNN.2008.2005605.

[72] C. Shang, S. Dash, M.F.M. Chowdhury, N. Mihindukulasooriya and A. Gliozzo, Taxonomy Construction of Unseen Domains via Graph-based Cross-Domain Knowledge Transfer, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Online, 2020, pp. 2198–2208. doi:10.18653/v1/2020.acl-main.199. https://aclanthology.org/2020.acl-main.199.

[73] J. Shen, Z. Shen, C. Xiong, C. Wang, K. Wang and J. Han, TaxoExpan: Self-supervised taxonomy expansion with position-enhanced graph neural network, in: *Proceedings of The Web Conference 2020*, 2020, pp. 486–497. https://arxiv.org/abs/2001.09522.

[74] L. Song, J. Zhang, L. Cheng, P. Zhou, T. Zhou and I. Li, NLPBench: Evaluating Large Language Models on Solving NLP Problems, *CoRR* **abs/2309.15630** (2023). doi:10.48550/arXiv.2309.15630.

[75] R. Speer, J. Chin and C. Havasi, ConceptNet 5.5: An Open Multilingual Graph of General Knowledge, 2018.

[76] R.C. Staudemeyer and E.R. Morris, Understanding LSTM–a tutorial into long short-term memory recurrent neural networks, *arXiv preprint arXiv:1909.09586* (2019). https://doi.org/10.48550/arXiv.1909.09586.

[77] K. Sun, Y.E. Xu, H. Zha, Y. Liu and X.L. Dong, Head-to-tail: How knowledgeable are large language models (llm)? AKA will llms replace knowledge graphs?, *arXiv preprint arXiv:2308.10168* (2023). https://arxiv.org/abs/2308.10168.

[78] K. Takeoka, K. Akimoto and M. Oyamada, Low-resource Taxonomy Enrichment with Pretrained Language Models, in: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 2021, pp. 2747–2758. doi:10.18653/v1/2021.emnlp-main.217. https://aclanthology.org/2021.emnlp-main.217.

[79] H. Tanev and A. Rotondi, Deftor at SemEval-2016 Task 14: Taxonomy enrichment using definition vectors, in: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, Association for Computational Linguistics, San Diego, California, 2016, pp. 1342–1345. doi:10.18653/v1/S16-1210. https://www.aclweb.org/anthology/S16-1210.

[80] R. Tang, X. Zhang, J. Lin and F. Ture, What Do Llamas Really Think? Revealing Preference Biases in Language Model Representations, *arXiv preprint arXiv:2311.18812* (2023). https://doi.org/10.48550/arXiv.2311.18812.

[81] A. Toral and R. Muñoz, A proposal to automatically build and maintain gazetteers for Named Entity Recognition by using Wikipedia, in: *Proceedings of the Workshop on NEW TEXT Wikis and blogs and other dynamic text sources*, 2006. https://aclanthology.org/W06-2809.

[82] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. Canton-Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P.S. Koura, M. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E.M. Smith, R. Subramanian, X.E. Tan, B. Tang, R. Taylor, A. Williams, J.X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov and T. Scialom, Llama 2: Open Foundation and Fine-Tuned Chat Models, *CoRR* **abs/2307.09288** (2023). doi:10.48550/arXiv.2307.09288.

[83] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. Canton-Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P.S. Koura, M. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E.M. Smith, R. Subramanian, X.E. Tan, B. Tang, R. Taylor, A. Williams, J.X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov and T. Scialom, LLaMA 2: Open Foundation and Fine-Tuned Chat Models, *CoRR* **abs/2307.09288** (2023). doi:10.48550/ARXIV.2307.09288. https://doi.org/10.48550/arXiv.2307.09288.

[84] P. Velardi, S. Faralli and R. Navigli, OntoLearn Reloaded: A Graph-Based Algorithm for Taxonomy Induction, *Computational Linguistics* **39**(3) (2013), 665–707. https://www.aclweb.org/anthology/J13-3007.

[85] I. Vulić and N. Mrkšić, Specialising Word Vectors for Lexical Entailment, in: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, M. Walker, H. Ji and A. Stent, eds, Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 1134–1145. doi:10.18653/v1/N18-1103. https://aclanthology.org/N18-1103.

[86] I. Vulić, D. Gerz, D. Kiela, F. Hill and A. Korhonen, HyperLex: A Large-Scale Evaluation of Graded Lexical Entailment, 2017.

[87] X. Wang, Y. Li, H. Wang and M. Lv, MKBQA: Question Answering over Knowledge Graph Based on Semantic Analysis and Priority Marking Method, *Applied Sciences* **13**(10) (2023). doi:10.3390/app13106104. https://www.mdpi.com/2076-3417/13/10/6104.

[88] Y. Wang, C. Liu, K. Chen, X. Wang and D. Zhao, SMASH: Improving SMAll Language Models' Few-SHot Ability with Prompt-Based Distillation, in: *Findings of the Association for Computational Linguistics: EMNLP 2022*, Y. Goldberg, Z. Kozareva and Y. Zhang, eds, Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 2022, pp. 6608–6619. doi:10.18653/v1/2022.findings-emnlp.492. https://aclanthology.org/2022.findings-emnlp.492.

[89] W. Xue, Y. Shen, W. Ren, J. Guo, S. Pu and W. Lu, Taxonomy Completion with Probabilistic Scorer via Box Embedding, *arXiv preprint arXiv:2305.11004* (2023).

[90] R. Ying, R. He, K. Chen, P. Eksombatchai, W.L. Hamilton and J. Leskovec, Graph Convolutional Neural Networks for Web-Scale Recommender Systems, in: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery; Data Mining*, KDD '18, ACM, 2018. doi:10.1145/3219819.3219890.

[91] Q. Zeng, J. Lin, W. Yu, J. Cleland-Huang and M. Jiang, Enhancing Taxonomy Completion with Concept Generation via Fusing Relational Representations, in: *KDD*, ACM, 2021, pp. 2104–2113. https://arxiv.org/pdf/2106.02974.

[92] J. Zhang, X. Song, Y. Zeng, J. Chen, J. Shen, Y. Mao and L. Li, Taxonomy completion via triplet matching network, in: *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35, 2021, pp. 4662–4670. http://www.arxiv.org/abs/2101.01896.

Appendix A. ChatGPT for Definition Generation and Automatic Error Analysis

We employed two distinct prompts, referenced as 9 and 10, with ChatGPT to create definitions for datasets that originally lacked them. Specifically, the MAG PSY and MAG CS datasets used in Taxonomy Enrichment, along with the ANT and HyperLex datasets for Lexical Entailment, did not have predefined definitions. To address this, we designed custom prompts for each dataset type. For hypernym prediction, the prompts are geared towards generating a definition for a single word, whereas for Lexical Entailment, definitions for two words are generated simultaneously to assist in disambiguation. The resulting definition statistics are provided in Table 13.

(9)

```
Write a definition for the word/phrase in one sentence.

Example:
Word: caddle
Definition: act as a caddie and carry clubs for a player

Word: eszopiclone 3 mg
Definition:
```

(10)

```
Write a definition for Word 1 and Word 2. Each definition should be
in one sentence. If a word is ambiguous, use the other word to
disambiguate it.

Example:
Word 1: depression
Word 2: melancholy
Definition 1: a mental state characterized by a pessimistic sense of
inadequacy and a despondent lack of activity
Definition 2: a constitutional tendency to be gloomy and depressed

Word 1: conflict
Word 2: disagreement
```

Dataset	Total	Generated with ChatGPT	From Wikidata
MAG PSY	23,156	12,823	10,333
MAG CS	29,484	5,714	23,770
ANT	5,933	5,933	-
HyperLex	2,307	2,307	-

Table 13
Statistics on definitions generated with ChatGPT for different tasks.

Example 11 shows how input was provided to ChatGPT for the purpose of automatically detecting error types in hypernym prediction. The corresponding output generated by the model is presented in Example 12. Additionally, Example 13 includes a prompt example used for automated error classification.

(11)

```
Here is the data in four columns separated with "\t":

topic: psychology input word: social construction of schizophrenia
true: psychiatry, predictions: psychosis, social, psychological,
and philosophical aspects, social psychology, psych, schizophrenia,
```

```
antipsychotic, psychotic, antipsychotic

topic: psychology input word: use heroin true: methadone maintenance,
predictions: heroin, opium tincture, heroin, opium, tincture, heroin,
opium, and morphine, man, heroin, drug addiction

...

topic: psychology input word: coordinated management of meaning
true: social psychology, predictions: communication disorders,
language and literacy, and social studies, language and literacy,
metaphor, metonymy, and politics, metonymy, metaphor

There is a topic, input word, correct hypernym (true), and model prediction
(predicted). Your task is to compare true answers and model predictions
and summarize what are the main error types of the model.
```

(12)

```
To summarize the main error types of the model's predictions compared
to the true hypernyms:

1. Overly Broad Predictions: The model often generates predictions that
encompass a broader concept than the true hypernym.
    For example:
        "medication effects" was predicted with terms like "drug effects",
        "drug interactions", and "medication", which are broader than
        the true hypernym "clinical psychology" and "psychiatry."

2. Underly Broad Predictions: Conversely, some predictions are too narrow
and fail to capture the broader concept represented by the true hypernym.
    For example:
        "schizoid personality" was predicted with terms like "personality
        disorders" and "personality theory," which are narrower than the
        true hypernym "psychosis."

3. Inclusion of Unrelated Terms: In some cases, the model includes terms
in its predictions that are not directly related to the input word or
true hypernym.
    For example:
        "pirisudanol" was predicted with terms like "dopamine receptor
        d1" and "dopamine receptor d2", which are unrelated to the true
        hypernym "psychiatry".

4. Repetition of Terms: The model occasionally repeats terms in its
predictions, which might indicate redundancy or lack of diversity
in its output.
    For example:
        "pirisudanol" was predicted with repeated terms like
        "piracetam" and "hydergine."
```

```
5. Inadequate Coverage of Concepts: Some input words and true
hypernyms receive predictions that lack comprehensive coverage of
related concepts.
    For example:
        "result category" was predicted with terms like "item
        response theory" and "qualitative research," which might not
        fully encapsulate the scope of the true hypernym "mathematics
        education."
```

(13)

```
You are a helpful linguist. Your task is to do error classification
for a dataset for hypernym prediction in a general domain. Here are
possible classes:

1. Overly Broad Predictions: The model often generates predictions
encompassing a broader concept than the true hypernym.
2. Underly Broad Predictions: Conversely, some predictions are too
narrow and fail to capture the broader concept represented by the
true hypernym.
3. Inaccurate Predictions: The model may predict words that are very
semantically close to the true hypernym, but struggles with fitting
into the exact wording
4. Conceptual Ambiguity: The model may struggle with ambiguous
(polysemantic/multivalued) input words or concepts, leading to
incorrect predictions.
5. Incorrect definitions: The model gets confused with the incorrect/
inaccurate definition retrieved from external sources

You will be given an input word/phrase, true hypernym, and
candidate hypernyms. Please, return a Python dict of error classes
{1: 1, 2: 5, 3: 1, ..., 100:3}) for all instances below:

id: 1, input word: parathyroid_hormone, true hypernym: hormone,
predicted: hormonal agent, hormon, hematopoietic growth factor,
growth factor of the blood, growth regulator, growth substance, growth
...
id: 100, input word: proofreader, true hypernym: printer, predicted:
reader, audience, audience member, spectator, viewer, listener,
listener-in, hearer, recipient, witness, watcher, observer
```

The prompt shown in Example 14 was used with ChatGPT to automatically evaluate TaxoLLaMA results, as manual analysis revealed that the gold standard answers in the MAG PSY and MAG CS datasets might not always be reliable. Consequently, ChatGPT was tasked with selecting between the dataset's gold standard answer and the model's predicted candidate.

(14)

```
    Here are the words in the psychological domain. Your task is to
    choose hypernym which is more relevant given two options.
```

```
    Answer 1 / 2 / both / none

Example:
social construction of schizophrenia
option 1: psychosis
option 2: psychiatry
Answer: 2

abdominal air sac
option 1: air sac
option 2: trachea
Answer:
```

Error Type	Descripton
Overly Broad Predictions	The model often generates predictions that encompass a broader concept than the true hypernym.
Underly Broad Predictions	Some predictions are too narrow and fail to capture the broader concept represented by the true hypernym
Inclusion of Unrelated Terms	In some cases, the model includes terms in its predictions that are not directly related to the input word or true hypernym.
Repetition of Terms	The model occasionally repeats terms in its predictions, which might indicate redundancy or lack of diversity in its output.
Inadequate Coverage of Concepts	Some input words and true hypernyms receive predictions that lack comprehensive coverage of related concepts
Semantic Shift	The model might exhibit errors related to semantic shift, where the predicted terms are semantically related to the input word but do not accurately reflect the intended meaning or context.
Conceptual Ambiguity	The model may struggle with ambiguous input words or concepts, leading to predictions that lack clarity or specificity.
Domain-Specific Knowledge	Errors may arise due to a lack of domain-specific knowledge or understanding of specialized terminology.
Cultural or Contextual Bias	The model's predictions may be influenced by cultural or contextual biases inherent in the training data. This could lead to inaccuracies, especially when dealing with topics or concepts that vary across cultures or contexts.
Incomplete Understanding of Relationships	The model may struggle to understand complex relationships between concepts, leading to inaccurate predictions.
Word Sense Disambiguation	Errors may occur due to difficulties in disambiguating between different senses of a word.
Knowledge Gap	The model's predictions may reflect gaps in its knowledge or understanding of certain concepts, resulting in inaccurate or incomplete responses.

Table 14

12 Error types made by TaxoLLaMA for hypernym prediction detected by ChatGPT.

Appendix B. Error Type Analysis

This section outlines the distribution of error types across different datasets for hypernym prediction, as detailed in Table 15. Furthermore, Table 16 provides an example of each error type, as classified by ChatGPT.

	1A: English	2A: Medical	2B: Music	1B: Italian	1C: Spanish	MAG-CS	MAG-PSY	Noun	Verb
Error 1	72.49%	93.75%	100.0%	54.69%	49.08%	66.48%	85.43%	81.45%	73.39%
Error 2	2.61%	0.00%	0.0%	10.03%	10.62%	5.40%	1.40%	4.10%	2.58%
Error 3	9.44%	1.25%	0.0%	27.83%	12.09%	13.21%	11.20%	6.05%	21.03%
Error 4	15.46%	5.00%	0.0%	7.44%	18.21%	10.94%	0.70%	6.64%	0.43%
Error 5	0.00%	0.00%	0.00%	0.00%	0.00%	3.98%	1.26%	1.76%	2.58%

Table 15

Errors Type Distribution across subset datasets for hypernym prediction: Hypernym Discovery and Taxonomy Enrichment.

Error Type	Input	True	Predicted
Overly Broad Predictions	weed	vascular plant	plant, plant life, vegetable, vegetation, vascular plant, phanerogam, green plant, green thing, plant of the class, plant of the division
Underly Broad Predictions	audio	auditory communication	signal, transmission, transmission signal, communication, communication signal, message, message signal, information, information signal, data, data signal, notification, notification signal, signaling
Inaccurate Predictions	unconventionality	unorthodoxy	unorthodoxiness, unorthodxy, unconformity, heterodoxy, dissidence, disobedience, disloyalty, dis, disaffection, dissent, disagreement
Conceptual Ambiguity	giant red paintbrush	indian paintbrush	paintbrushrush, paint brush, larkspur, delphinium, liliaceous plant, liliid monocot, lilioid
Incorrect definitions	advance directive status	documentation	advance directive, directive, decision support system, clinical decision support system, health information technology

Definition: A do-not-resuscitate order (DNR), also known as Do Not Attempt Resuscitation (DNAR), Do Not Attempt Cardiopulmonary Resuscitation (DNACPR)

Table 16

Examples for each Error type made by TaxoLLaMA for hypernym prediction detected by ChatGPT.

Appendix C. Distribution Visualization for Taxonomy Enrichment

In this section, we explore the distribution of ground truth and model predictions within the Sentence-Bert embedding space [70]. To ensure the results were not tied to a specific initialization, we performed two model runs with different seeds. We then projected the predicted candidates and ground truth hypernyms into the SentenceBert embedding space. For better visualization, we first reduced the embeddings to 50

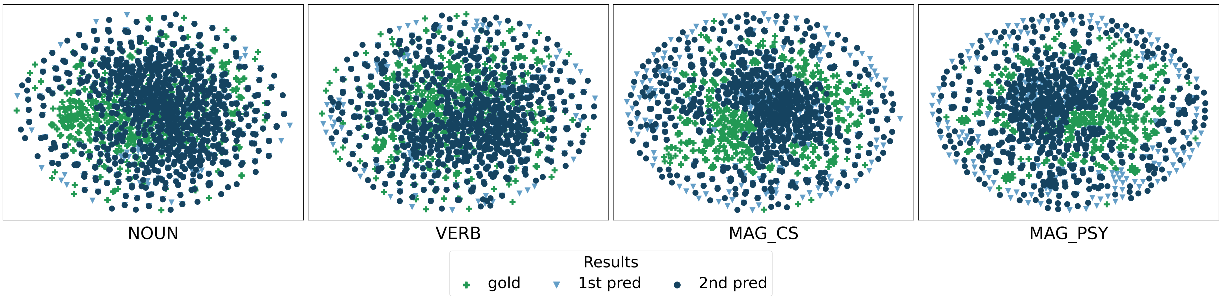


Fig. 6. t-SNE plot of distributions of ground truth nodes and predicted nodes for taxonomy enrichment tasks. Each point represents a node, embedded with SentenceBert. Color represents ground truth or model predictions (we ran 2 predictions with different seeds)

dimensions using Principal Component Analysis (PCA), followed by t-SNE to condense the data into a two-dimensional space.

Figure 6 uncovers differences in how WordNet and the MAG subsets (MAG_CS and MAG_PSY) are represented in the embedding space. For WordNet, there is considerable overlap between the model's predictions and the gold standard, with only a few exceptions, likely linked to lower-ranked candidates. In contrast, the MAG subsets form two distinct clusters that barely overlap, indicating a notable divergence between the predicted and true hypernyms. Moreover, the MAG subsets contain more outliers, suggesting that the model may have missed the correct hypernym sense entirely in several instances. These observations could be influenced by the SentenceBert model's limitations, especially when dealing with concepts that are not well-represented in the training data.

Appendix D. Hyperlex Correlation Analysis

We also evaluated correlations using traditional methods for both test sets, as shown in Figure 7. A clear pattern emerges when a linear regression line is added to the data points, though this pattern is heavily influenced by outliers, particularly in the Random set. This finding is consistent with those from taxonomy construction, where the model also faces difficulties in accurately processing middle nodes or pairs with moderate entailment strength.

When evaluating gold scores in the range of 2 to 8, the Random set shows no discernible trend, underscoring the model's inconsistency in this area. The Lexical set, on the other hand, exhibits a slightly more defined trend within the same range. Nonetheless, in both sets, pairs with either strong or minimal entailment are more reliably categorized. This differentiation significantly enhances the overall correlation, contributing to an encouraging correlation score.

Appendix E. Hyperparameter motivation

Our investigation revealed the model's pronounced sensitivity to both learning rate and scheduler settings. In the initial experiments, the successful application of a high learning rate was largely attributed to the LORA adapter, which subtly adjusts weights without causing major disruptions. However, when engaging in full model fine-tuning, we encountered significant instability, with the model oscillating between overfitting and underfitting, underscoring the need for refined hyperparameter tuning. Additionally, implementing 4-bit quantization requires careful calibration of the learning rate, as this compression method significantly alters the weight distribution, making it necessary to employ strategies that effectively restore the model's knowledge.

During fine-tuning, we chose a smaller batch size to better align the model with datasets that often have limited samples. However, increasing the learning rate and batch size did not enhance performance, likely

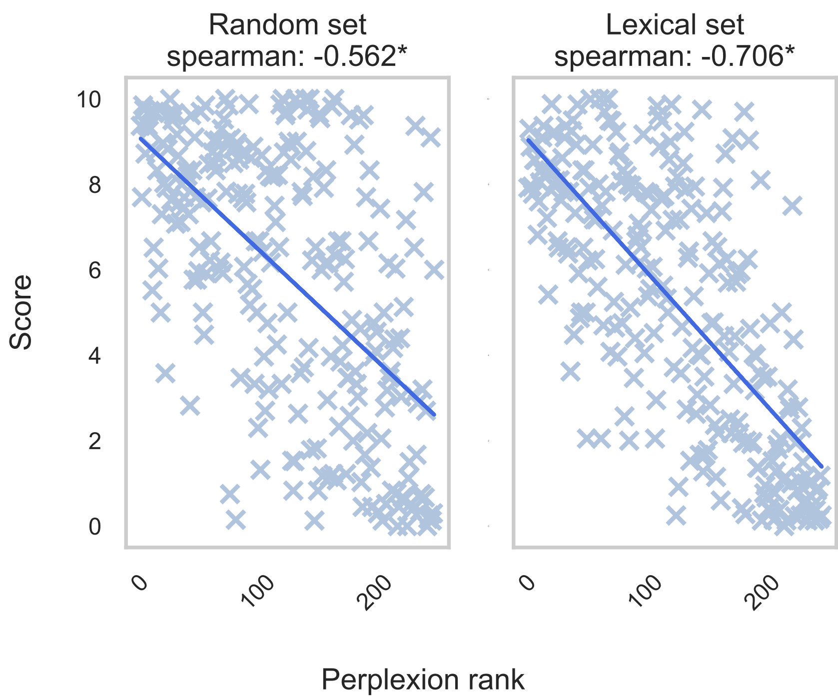


Fig. 7. Correlation plot of the perplexion ranks with the annotator's score on Hyperlex test sets. The line over the dots is a trend found with linear regression. * shows that correlation has a p-value lower than $1e^{-4}$.

due to the model having fewer steps to adapt to domain-specific features. This was not the case during WordNet pre-training, where different trends were observed.

In contrast to certain instruction tuning strategies, our method does not calculate loss based on the instruction itself but rather focuses exclusively on the target tokens.

Our experiments were carried out on Nvidia A100 or Quadro RTX 8000 GPUs. Pre-training for both TaxoLLaMA and TaxoLLaMA$_{\text{bench}}$ took around 6 GPU hours, while TaxoLLaMA$_{\text{verb}}$ required less than 1 hour. Fine-tuning the MAG subsets took 5 GPU hours due to the extended definitions, whereas other datasets were fine-tuned in under an hour.