Enhancing Small Open-Source Language Models for Ontology Generation through Metric-Guided Continual Pretraining

Miquel Canal-Esteve

Research Group of Language Processing and Information System, University of Alicante, Spain

Abstract

Ontology development requires expert knowledge and structural precision. While Large Language Models (LLMs) show promise for ontology tasks, small open-source models like Llama 3.2-1B still lack strong semantic and structural understanding. We propose a two-phase approach: continual pretraining on high-quality ontology datasets, guided by two frameworks—one for semantic metrics and another for lexical-structural metrics. We pretrained Llama 3.2-1B using semantic-based high-quality subsets and evaluated improvements through manual and structural analyses. Results show small, high-quality subsets yield rapid gains, while larger, diverse datasets improve long-term performance. Since semantic metrics need complete ontologies, ORI remains key for evaluating fragments. Future work will apply instruction- and fine-tuning for specialized tasks such as those in the LLMs4OL benchmark or for generating structured resources across domains using ontology-based methods. This work shows that thoughtful data selection and continual pretraining can push small LLMs toward expert-level ontology generation.

Keywords

Ontology generation, continual pretraining, semantic metrics, large language models

1. Justification of the Proposed Research

Ontology development is a demanding and time-intensive process that requires expert knowledge, precise vocabulary design, and careful structural coherence [1, 2]. Despite progress in ontology engineering methodologies, manual development remains common [3, 4], creating a growing need for automated approaches [5]. Large Language Models (LLMs) have emerged as promising tools for supporting ontology-related tasks such as matching, enrichment, and generating structured knowledge from textual data [6, 7].

Among these, small, open-source LLMs like Llama 3.2-1B [8], Gemma 1.1B [9], and Pythia-1B [10] stand out for their computational efficiency, accessibility, and adaptability. However, exploratory research shows that these models lack strong ontology or semantic knowledge. When tested on the ontology-to-ontology generation task — where a model receives an ontology fragment and is expected to continue it coherently — they show significant weaknesses in lexical richness, structural accuracy, and triple generation, making frequent mistakes. This task serves as a quick diagnostic of the model's underlying ontology-handling capacity and reveals critical gaps to address.

The proposed research aims to improve small, open-source LLMs' capacity for ontology understanding and generation by applying continual pretraining on high-quality semantic datasets, guided by combined semantic, lexical, and structural metrics. Building on this foundation, the work will later apply instruction-tuning to specialize the models for specific ontology tasks as defined in the LLMs4OL [7] benchmark, such as term typing, taxonomy discovery, and relation extraction.

These tasks aim to advance small, open-source LLMs in ontology engineering while also enabling automated ontology creation across diverse domains, like education, biomedicine, and beyond.

Doctoral Symposium on Natural Language Processing, 25 September 2025, Zaragoza, Spain.

mikel.canal@ua.es (M. Canal-Esteve)

0009-0006-8022-5534 (M. Canal-Esteve)

2. Related Work

Ontology-to-ontology generation remains a largely untapped area, especially when considering the capabilities of large language models (LLMs). While most prior research focuses on applying LLMs to tasks like ontology refinement, enrichment, or generation from unstructured text sources [6, 7, 11], these works primarily provide the groundwork upon which this study builds.

Recent research highlights the diverse roles LLMs can play in ontology engineering. For instance, Zhao et al. [12] incorporate OntoClean principles to improve refinement; Toro et al. [13] leverage retrieval-augmented generation (RAG) in DRAGON-AI for dynamic construction; Fathallah et al. [14] address structured translation using NeOn-GPT; Zhang et al. [15] develop conversational approaches with OntoChat; He et al. [16] apply deep learning for ontology completion in DeepOnto; and Mukanova et al. [17] use LLMs for enrichment tasks. Yet, none of these approaches directly tackle the challenge of autonomously generating new ontologies from partial or incomplete inputs.

Additional research on text-to-ontology generation also informs our approach. Babaei et al. [7] break down the generation process into subtasks and emphasize the importance of fine-tuning; Saeedizade et al. [11] guide progressive ontology construction through competency questions; and Da Silva et al. [18] demonstrate how few-shot prompting can enhance generation outcomes. By contrast, our work adopts continual pretraining to help the model internalize domain-relevant patterns and knowledge prior to task-specific fine-tuning, thus reducing dependence on prompt-driven examples.

Moreover, the importance of data cleaning for improving model performance is well established across the literature [19, 20], with several studies introducing sophisticated filtering methods during preprocessing [21, 22, 23]. However, these methods are predominantly designed for unstructured text, and to date, no standardized methodology exists for selecting high-quality data specifically for continual pretraining in the context of ontologies. Our study addresses this gap by proposing a systematic approach to identify and leverage high-quality ontological data for improving LLM performance.

3. Description of the Proposed Research

This research is structured in two main phases. First, we focus on improving the models' general semantic and ontology knowledge through continual pretraining on ontology datasets. For this, we have developed two dedicated repositories to measure the quality of datasets: one for computing semantic metrics [1] (covering classes, taxonomic and non-taxonomic relations) and another for computing lexical and structural metrics [2] (focused on vocabulary usage and structural patterns). The goal is to combine these into a global metric that can evaluate the quality of both the datasets and the outputs generated by the model, helping to identify high-quality subsets — since it is well known that less data of higher quality is more effective for training. Some experiments have already been carried out in this direction.

Second, we will apply instruction-tuning and fine-tuning to adapt the continually pretrained models for specific, high-value ontology tasks, such as those defined in the LLMs4OL [7] benchmark — including term typing, type taxonomy discovery, and type non-taxonomic relation extraction. Together, these efforts aim to systematically enhance small, open-source LLMs' ability to handle advanced ontology engineering challenges and structured knowledge tasks more effectively than base models. This tasks can help to automatize the creation of didacic material based on ontologies.

4. Methodology

This section details the methodological framework designed to improve small open-source language models for ontology generation. We combine curated ontology repositories, carefully designed semantic and lexical-structural metrics, and a continual pretraining strategy. By integrating dataset selection, metric-driven evaluation, and robust training configurations, our approach systematically enhances

[1] https://github.com/miquelcanalesteve/LLM4Onto

[2] https://github.com/miquelcanalesteve/ontology-metrics-pretraining

the model's semantic, lexical, and structural capabilities. Below, we describe the ontology sources, the metric systems, the segmentation strategies, the evaluation framework, and the pretraining setup.

4.1. Ontology Repository

We base our methodology on an ontology repository that provides structured knowledge for pretraining. These repositories include ontologies of varying size, completeness, and semantic richness, requiring additional filtering before use.

For this study, we selected DBpedia Archivo[3], a widely used repository of ontologies across diverse domains [24]. Its files are provided in TTL (Turtle) format, a popular and human-readable RDF serialization, making it ideal for evaluating how quality-based dataset selection affects model performance.

Our dataset, downloaded on July 15, 2024, includes 1,766 ontologies totaling 71 million triples, ranging from small sets under 10 triples to large ontologies exceeding 10 million.

4.2. Lexical and Structural Ontology Metrics

This section introduces the text-based metrics designed to quantify vocabulary usage, lexical richness, and structural variability in ontology files.

To evaluate an ontology, we assess its raw text representation—whether it comes from an existing dataset or is generated by a model—using lightweight text-based metrics inspired by Palomar et al. [19]. These metrics capture vocabulary use and structural diversity without requiring reasoning or formal parsing. We then aggregate them into the Ontology Reference Index (ORI), drawing on concepts from [25], to support data ranking and performance evaluation.

Vocabulary-specific density. Average number of predefined vocabulary terms per non-empty line (dependent on typical one-relation-per-line format):

$$V_{\text{den}} = \frac{1}{L} \sum_{i=1}^{L} c_i$$

where L is the number of non-empty lines, and c_i is the number of vocabulary terms detected in line i. The vocabulary is a predefined set of ontology modelling terms commonly used across structured knowledge representations, including those from RDF, RDFS, OWL, and XSD (the full vocabulary is available in the repository). Terms inside quoted literals are excluded.

Vocabulary-specific diversity. Proportion of vocabulary terms that appear at least once in the file:

$$V_{\text{div}} = \frac{|V_{\text{doc}}|}{|V_{\text{spec}}|}$$

where $V_{\text{doc}} \subseteq V_{\text{spec}}$ is the set of vocabulary terms found in the file, and V_{spec} is the same vocabulary used for V_{den}. A higher value indicates broader use of available modeling constructs.

Logical block uniqueness ratio (LBUR). Fraction of unique logical blocks in the ontology:

$$\text{LBUR} = \frac{|\text{unique\_blocks}|}{|\text{blocks}|}$$

Logical blocks are defined as minimal self-contained RDF/OWL units, starting from a subject and continuing until the terminating period. These typically include class declarations, property assertions, or grouped triples.

[3] https://archivo.dbpedia.org/

Line uniqueness ratio (LUR). Fraction of unique non-empty lines:

$$\text{LUR} = \frac{|\text{unique\_nonempty\_lines}|}{|\text{nonempty\_lines}|}$$

This metric captures surface-level textual redundancy, regardless of line type (structural, directive, or annotation).

Brunet Index (BI). Lexical richness index:

$$\text{BI} = N^{V^{-0.165}}$$

where N is the total number of word tokens and V is the number of unique word types. Composite terms (e.g., prefix-based identifiers) are tokenized accordingly. Lower values indicate greater lexical diversity.

Ontology Reference Index (ORI) and Evaluation The Ontology Reference Index (ORI) provides a weighted measure of an ontology's alignment with an idealized reference, which aggregates the best observed values for each of the five previously defined metrics. This reference does not represent any single ontology but instead reflects the per-metric maxima identified across the dataset.

The computation normalizes all metric values using min-max scaling. Because lower Brunet Index values indicate better lexical diversity, the method inverts this metric using $1 - n(BI)$, where $n(BI)$ denotes the normalized value. The ORI score is then calculated as:

$$ORI = \sum_{m}^{m \in M} \omega(m) * N(m)$$

where $M = \{V_{den}, V_{div}, LBUR, LUR, BI\}$, and

$$N(m) = \begin{cases} n(m), & \text{if } m \neq BI \\ 1 - n(m), & \text{if } m = BI \end{cases}$$

The weight ω assigned to each metric reflects the performance gap between the base model (Llama 3.2-1B) and the top-performing ontology for that metric. For the Brunet Index, the method computes this ratio inversely (base / best) to maintain consistency with its inverted interpretation. The procedure then normalises these gains to derive the final weights, which appear in Table 1.

	V_{den}	V_{div}	**LBUR**	**LUR**	**BI**
Llama 3.2-1B	0.500	0.035	0.955	0.738	16.11
Top-1 dataset	1.257	0.622	1	1	4.382
Gain	2.514	17.744	1.048	1.345	3.679
Weights	**0.096**	**0.673**	**0.040**	**0.051**	**0.140**

Table 1
Weight calculation for ORI. Gains are computed as the ratio between base and top metric values (inverted for Brunet) and normalised to produce final weights.

To estimate base model values, we sampled 12 ontology fragments of 150 tokens each and generated 6 completions of 450 tokens per fragment. The generation used the following configuration: `do_sample=True`, `top_k=50`, `top_p=0.95`, and `temperature=0.7`. The trained models followed the same ontology completion protocol.

4.3. Semantic Metrics

To evaluate the structural quality of an ontology, we propose lightweight complexity-based metrics inspired by Tello et al. [26] and Gutiérrez et al. [27]. These metrics quantify the richness and density of the ontology without requiring reasoning or formal entailment, making them scalable for large repositories. We then aggregate them into a unified quality score to support dataset filtering and model evaluation.

Average Subclasses per Class (SC). Average number of subclasses per class, reflecting the hierarchical depth and granularity of the ontology taxonomy:

$$SC = \frac{\sum_{i=1}^{c} s(i)}{c}$$

where c is the total number of classes and $s(i)$ is the number of subclasses of class i.

Average Non-Taxonomic Relations per Class (NTR). Average number of non-taxonomic relationships per class, indicating the density of semantic links beyond simple hierarchies:

$$NTR = \frac{\sum_{i=1}^{c} r_{\text{not}}(i)}{c}$$

where $r_{\text{not}}(i)$ is the number of non-taxonomic relationships attached to class i.

Property Density (PD). Average number of attributes and non-taxonomic relations per class, serving as a proxy for schema richness and information density:

$$PD = \frac{\sum_{i=1}^{c} (n_{\text{att}}(i) + r_{\text{not}}(i))}{c}$$

where $n_{\text{att}}(i)$ denotes the number of data properties (attributes) of class i.

To consolidate these aspects into a unified quality score QS, we normalize the three metrics using min-max scaling and compute:

$$QS = n(PD) + n(SC) + n(NTR)$$

These metrics are computed using the rdflib Python library, providing an efficient and reproducible basis for ontology quality analysis.

4.3.1. Segmentation of Datasets

While the segmentation approach described here is applied using the QS metric, the same logic could be extended to the Ontology Reference Index (ORI) or other metric, allowing future work to explore dataset splits that prioritize lexical and structural quality alongside semantic complexity. For this study, however, segmentation is based solely on QS, which focuses on semantic richness, density, and hierarchy.

To segment the dataset, we first compute the token count for each ontology. This allows us to define partitions based on token distribution, ensuring that different subsets capture varying levels of quality and diversity (i.e., more ontologies lead to greater diversity). While segmentation can be done in multiple ways—by quartiles, deciles, halves, or other thresholds—we adopt three specific strategies:

1. **Q1 (Prioritizing Quality)**: Ontologies are ranked by QS, and those with the highest scores are selected until reaching at least 25% of the total tokens. Since selection is done without truncation, the last ontology added may slightly exceed this threshold. In our case, this resulted in 31% of the total tokens.

2. **Q1,2 (Quality + Diversity)**: Ontologies are again ranked by QS, and selection continues until reaching at least 50% of the total tokens. This strategy balances quality and diversity while ensuring that no ontology is arbitrarily truncated.
3. **Q1-4 (Full Dataset)**: This set includes all available ontologies, covering the entire range of quality levels and structural complexities. It serves as a baseline to assess the impact of training on the full, unfiltered dataset.

This segmentation enables a systematic assessment of how training on subsets with varying semantic quality affects model performance. Table 3 summarizes the selected datasets, showing the average values and standard deviations for each key quality metric.

4.4. Manual Evaluation

The manual evaluation framework is based on da Silva et al. [18], which categorizes errors into syntactic, semantic, and structural issues to comprehensively assess ontology quality. Additional criteria follow Chen et al. [28] to address ambiguity and redundancy, and Xu et al. [29] to capture text repetition. Errors include syntactic violations (e.g., missing delimiters), triplet repetition, text repetition within comments or literals, semantic redundancy, ambiguity between entities, semantic contradictions (e.g., conflicting OWL types), and vocabulary misuse involving incorrect ontology terms. A complete guide for the evaluation is available in the repository [4].

To quantify performance, we compute the mean error rate per triple across categories, following da Silva et al. [18]. The evaluation uses unseen ontology fragments drawn from diverse repositories such as AGRO[5], EDAM[6], MDS[7], and SWEET[8], covering domains like biology, spatial data, and agriculture. Each fragment (150 tokens) was randomly sampled and generated six times using the Hugging Face library with `do_sample=True`, `top_k=50`, `top_p=0.95`, and `temperature=0.7`, ensuring robust and unbiased measurement of generalization capabilities.

4.5. Pretraining LLM

For continual pretraining, we used the Llama 3.2-1B model [8], chosen for its compact yet expressive 1.2 billion parameter architecture, which balances adaptability and computational efficiency. The TTL ontologies were processed as plain text, allowing standard NLP tokenization without specialized parsing, resulting in 1.25 billion tokens across all subsets. To scale training effectively, we applied a Distributed Data Parallel (DDP) strategy [30], distributing parameters and gradients across four NVIDIA A100 GPUs (40 GB each), with gradient accumulation and checkpointing to optimize batch size. Each dataset subset was pretrained for two epochs, as longer runs showed no additional gains.

5. Experiments

The results reported in Table 2 use quartiles selected based on semantic quality metrics, which guided the segmentation of the dataset into high-quality (**Q1**), top-half (**Q1,2**), and full (**Q1...4**) subsets. While the same segmentation logic could, in principle, be applied using the Ontology Reference Index (ORI) to emphasize lexical and structural quality, in this study it was only tested with the QS metric, which focuses on semantic richness, density, and hierarchy.

Importantly, QS cannot be applied to model-generated outputs because these are only partial ontology fragments, and the `rdflib` library requires complete, parsable ontology structures to compute these metrics, whereas ORI remains applicable because it operates directly over the raw text representation.

[4] https://github.com/miquelcanalesteve/LLM4Onto/tree/main/results
[5] https://bioportal.bioontology.org/ontologies/AGRO
[6] https://bioportal.bioontology.org/ontologies/EDAM
[7] https://matportal.org/ontologies/MDS
[8] https://earthportal.eu/ontologies/SWEET

Model	Ep	Syn	Rep	TxtRep	Red	Amb	Sem	Voc	Total
Base	-	0.030	0.307	0.094	0.021	**0.007**	**0.000**	0.000	0.066
Q1	**1**	**0.006**	**0.044**	0.020	0.013	**0.007**	0.006	0.018	**0.016**
Q1	2	0.023	0.063	0.025	0.022	0.010	0.023	0.004	0.024
Q1,2	1	0.009	0.101	0.008	0.012	0.019	0.003	0.013	0.024
Q1,2	2	0.082	0.130	0.024	**0.011**	0.021	0.009	0.000	0.040
Q1...4	1	0.025	0.103	**0.005**	0.018	0.021	0.001	0.000	0.025
Q1...4	2	0.021	0.072	0.009	0.023	0.015	0.007	0.001	0.025

Table 2
Mean error rates per triple for different training configurations. Abbreviations: Syn = Syntactic, Rep = Repetition, TxtRep = Text Repetition, Red = Redundancy, Amb = Ambiguity, Sem = Semantic, Voc = Vocabulary. Error values indicate the proportion of affected triples relative to the total generated triples. The "Total" column represents the mean error rate across all error categories.

The base Llama 3.2-1B model shows a high total error rate of **6.6%**, mainly driven by repetition errors (**30.7%**) and syntactic issues (**3.0%**). Semantic and vocabulary-specific errors are almost negligible in the base outputs, reflecting structurally shallow generations. Pretraining on the high-quality subset (**Q1**) for one epoch sharply reduces the total error rate to **1.6%**, with substantial improvements in repetition (**4.4%**) and syntactic errors (**0.6%**). A second epoch on **Q1** slightly increases total errors to **2.4%**, suggesting diminishing returns or mild overfitting.

Expanding the training to larger subsets, such as **Q1,2** or the full dataset (**Q1...4**), stabilizes error rates between **2.4%** and **2.5%**, with the best redundancy and text repetition reduction achieved under the **Q1,2 (2 epochs)** and **Q1...4 (1 epoch)** settings. These configurations show that simply increasing dataset size or epochs does not linearly improve performance, making it crucial to calibrate training parameters carefully.

Model	Ep.	V_{den}	V_{div}	LBUR	LUR	BI	ORI
Base	-	0.500	0.035	0.955	0.738	16.11	0.252
Q1	1	0.584	0.050	0.978	**0.902**	14.18	0.293
Q1	2	0.599	0.056	0.963	0.856	14.23	0.295
Q1,2	1	0.607	0.054	0.988	0.863	**14.02**	0.299
Q1,2	2	0.609	0.050	0.990	0.870	14.03	0.296
Q1...4	1	0.627	0.052	0.989	0.875	14.08	0.300
Q1...4	**2**	**0.631**	**0.057**	**0.994**	0.891	14.05	**0.308**

Table 3
Evaluation of the base model and continually pretrained models on lexical and structural metrics. Reported are vocabulary density and diversity, logical block and line uniqueness, Brunet Index, and aggregated ORI scores, with relative improvements over the base model (in parentheses).

Table 3 summarizes the lexical and structural quality metrics, aggregated into the Ontology Reference Index (ORI). The base model starts with an ORI of **0.252**, reflecting low scores in vocabulary density (**0.500**), diversity (**0.035**), and line uniqueness (**0.738**), alongside a high Brunet Index (**16.11**), indicating limited lexical richness. Pretraining on **Q1** improves ORI to **0.293**, while the best overall performance is achieved after two epochs on the full dataset (**Q1...4**), reaching an ORI of **0.308**—a **22%** improvement over the base. These gains confirm that continual pretraining fosters improvements not only in reducing surface-level errors but also in enhancing deeper lexical and structural qualities.

6. Conclusions and Future Work

Overall, the results demonstrate that continual pretraining meaningfully boosts the model's ability to generate coherent, semantically aligned, and structurally rich ontologies. While small, high-quality subsets like **Q1** enable rapid improvements, broader datasets like **Q1...4** maximize long-term structural

gains, provided training configurations are carefully balanced to avoid performance plateaus. These findings highlight the need to rethink data selection strategies: although this study segmented data using semantic metrics, future work should explore integrating lexical and structural dimensions into a combined metric. Such a mixed metric could help isolate subsets that offer the best balance between semantic depth, lexical richness, and structural complexity, potentially driving even more robust model improvements.

Additionally, the evaluation framework itself presents an opportunity for advancement. The current manual assessment, while informative, is labor-intensive and limits scalability; developing an automated evaluation pipeline would not only streamline the process but also enhance reproducibility and allow finer-grained analysis across larger test sets. Looking ahead, the next research phase will apply instruction tuning and task-specific fine-tuning, aligning pretrained models with specialized ontology tasks such as those outlined in the LLMs4OL [7] benchmark, as well as expanding applications across diverse domains like education and biomedicine. Together, these steps aim to move small, open-source LLMs beyond general improvements toward expert-level performance in key ontology engineering applications.

Declaration on Generative AI

During the preparation of this work, the authora used ChatGPT in order to: Grammar and spelling check, Paraphrase, translate and reword. After using this tool/service, the authors reviewed and edited the content as needed and takes full responsibility for the publication's content.

References

[1] M. Fernández-López, A. Gómez-Pérez, N. Juristo, Methontology: from ontological art towards ontological engineering (1997).

[2] M. Poveda-Villalón, A. Fernández-Izquierdo, M. Fernández-López, R. García-Castro, Lot: An industrial oriented ontology engineering framework, Engineering Applications of Artificial Intelligence 111 (2022) 104755.

[3] P. Lambrix, R. Armiento, H. Li, O. Hartig, M. Abd Nikooie Pour, Y. Li, The materials design ontology, Semantic Web (2024) 1–35.

[4] Y. Lu, G. Song, P. Li, N. Wang, Development of an ontology for construction carbon emission tracking and evaluation, Journal of Cleaner Production 443 (2024) 141170.

[5] A. Amalki, K. Tatane, A. Bouzit, Deep learning-driven ontology learning: A systematic mapping study, Engineering, Technology & Applied Science Research 15 (2025) 20085–20094.

[6] R. Du, H. An, K. Wang, W. Liu, A short review for ontology learning: Stride to large language models trend, arXiv preprint arXiv:2404.14991 (2024).

[7] H. Babaei Giglou, J. D'Souza, S. Auer, Llms4ol: Large language models for ontology learning, in: International Semantic Web Conference, Springer, 2023, pp. 408–427.

[8] Llama, Model cards and prompt formats - llama 3.2, https://www.llama.com/docs/model-cards-and-prompt-formats/llama3_2/, 2024. Accessed: 2025-03-04.

[9] G. Team, A. Kamath, J. Ferret, S. Pathak, N. Vieillard, R. Merhej, S. Perrin, T. Matejovicova, A. Ramé, M. Rivière, et al., Gemma 3 technical report, arXiv preprint arXiv:2503.19786 (2025).

[10] S. Biderman, H. Schoelkopf, Q. G. Anthony, H. Bradley, K. O'Brien, E. Hallahan, M. A. Khan, S. Purohit, U. S. Prashanth, E. Raff, et al., Pythia: A suite for analyzing large language models across training and scaling, in: International Conference on Machine Learning, PMLR, 2023, pp. 2397–2430.

[11] M. J. Saeedizade, E. Blomqvist, Navigating ontology development with large language models, in: European Semantic Web Conference, Springer, 2024, pp. 143–161.

[12] Y. Zhao, N. Vetter, K. Aryan, Using large language models for ontoclean-based ontology refinement, arXiv preprint arXiv:2403.15864 (2024).

[13] S. Toro, A. V. Anagnostopoulos, S. M. Bello, K. Blumberg, R. Cameron, L. Carmody, A. D. Diehl, D. M. Dooley, W. D. Duncan, P. Fey, et al., Dynamic retrieval augmented generation of ontologies using artificial intelligence (dragon-ai), Journal of Biomedical Semantics 15 (2024) 19.
[14] N. Fathallah, A. Das, S. D. Giorgis, A. Poltronieri, P. Haase, L. Kovriguina, Neon-gpt: a large language model-powered pipeline for ontology learning, in: European Semantic Web Conference, Springer, 2024, pp. 36–50.
[15] B. Zhang, V. A. Carriero, K. Schreiberhuber, S. Tsaneva, L. S. González, J. Kim, J. de Berardinis, Ontochat: a framework for conversational ontology engineering using language models, in: European Semantic Web Conference, Springer, 2024, pp. 102–121.
[16] Y. He, J. Chen, H. Dong, I. Horrocks, C. Allocca, T. Kim, B. Sapkota, Deeponto: A python package for ontology engineering with deep learning, Semantic Web 15 (2024) 1991–2004.
[17] A. Mukanova, M. Milosz, A. Dauletkaliyeva, A. Nazyrova, G. Yelibayeva, D. Kuzin, L. Kussepova, Llm-powered natural language text processing for ontology enrichment., Applied Sciences (2076-3417) 14 (2024).
[18] L. M. V. da Silva, A. Kocher, F. Gehlhoff, A. Fay, On the use of large language models to generate capability ontologies, in: 2024 IEEE 29th International Conference on Emerging Technologies and Factory Automation (ETFA), IEEE, 2024, pp. 1–8.
[19] J. Palomar-Giner, J. J. Saiz, F. Espuña, M. Mina, S. Da Dalt, J. Llop, M. Ostendorff, P. O. Suarez, G. Rehm, A. Gonzalez-Agirre, et al., A curated catalog: Rethinking the extraction of pretraining corpora for mid-resourced languages, in: Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), 2024, pp. 335–349.
[20] P. J. O. Suárez, B. Sagot, L. Romary, Asynchronous pipeline for processing huge corpora on medium to low resource infrastructures, in: 7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7), Leibniz-Institut für Deutsche Sprache, 2019.
[21] Z. Chen, Y. Liu, L. Chen, S. Zhu, M. Wu, K. Yu, Opal: Ontology-aware pretrained language model for end-to-end task-oriented dialogue, Transactions of the Association for Computational Linguistics 11 (2023) 68–84.
[22] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, arXiv preprint arXiv:1911.02116 (2019).
[23] S. Kudugunta, I. Caswell, B. Zhang, X. Garcia, D. Xin, A. Kusupati, R. Stella, A. Bapna, O. Firat, Madlad-400: A multilingual and document-level large audited dataset, Advances in Neural Information Processing Systems 36 (2023) 67284–67296.
[24] J. Frey, D. Streitmatter, F. Götz, S. Hellmann, N. Arndt, Dbpedia archivo: a web-scale interface for ontology archiving under consumer-oriented aspects, Semantic Systems. In the Era of Knowledge Graphs 12378 (2020) 19.
[25] H. Alani, C. Brewster, Metrics for ranking ontologies, in: Proceedings of the Evaluating Ontologies for the Web Workshop (EON2006), 15th International World Wide Web Conference, EON Workshop, Edinburgh, Scotland, 2006.
[26] A. J. L. Tello, Métrica de idoneidad de ontologías, Ph.D. thesis, Universidad de Extremadura, 2002.
[27] Y. Gutierrez, D. Tomas, I. Moreno, Developing an ontology schema for enriching and linking digital media assets, Future Generation Computer Systems 101 (2019) 381–397.
[28] H. Chen, G. Cao, J. Chen, J. Ding, A practical framework for evaluating the quality of knowledge graph, in: Knowledge Graph and Semantic Computing: Knowledge Computing and Language Understanding: 4th China Conference, CCKS 2019, Hangzhou, China, August 24–27, 2019, Revised Selected Papers 4, Springer, 2019, pp. 111–122.
[29] J. Xu, X. Liu, J. Yan, D. Cai, H. Li, J. Li, Learning to break the loop: Analyzing and mitigating repetitions for neural text generation, Advances in Neural Information Processing Systems 35 (2022) 3082–3095.
[30] J. Duan, S. Zhang, Z. Wang, L. Jiang, W. Qu, Q. Hu, G. Wang, Q. Weng, H. Yan, X. Zhang, et al., Efficient training of large language models on distributed infrastructures: a survey, arXiv preprint

arXiv:2407.20018 (2024).