Classifying domain entities into top-level ontology concepts using informal definitions: A multi-label, multi-resource, and multi-language study

Alcides Lopes [a,*], Joel Carbonera [a] and Mara Abel [a]
[a] *Institute of Informatics, Universidade Federal do Rio Grande do Sul, Porto Alegre, 91501970, Brazil*

Abstract.
Identifying which top-level ontology concept a domain entity specializes in is laborious and time-consuming because it is usually performed manually and requires a high level of expertise in both the target domain and ontology engineering. This study proposes an approach to classifying domain entities into top-level ontology concepts using informal definitions. We explore the hypothesis that the informal definitions represent semantic information that allows domain entities to be related to top-level ontology concepts. Our research spans multiple languages and leverages diverse resources, making it a comprehensive multi-language and multi-resource investigation. Leveraging state-of-the-art language models, we propose two distinct classification pipelines. The first pipeline fine-tunes pre-trained language models to our specific task, while the second employs these models to generate embeddings for classical machine learning classifiers. This dual approach allows us to navigate the complexities of the distributional hypothesis effectively. We extract a rich multi-label dataset from the alignment of OntoWordNet with BabelNet, expanding our analysis across 291 languages and various semantic resources, such as WordNet and Wikipedia. This expansive dataset enables us to validate our hypothesis and investigate the nuances of domain entity classification in a multi-label, multi-language, and multi-resource context. Our experiments explored the performance of different training methodologies and state-of-the-art language models. From that, our best results were achieved using the K-Nearest Neighbor (KNN) approach fed by informal definition embedding from the Mistral7B language model. This result, in addition to validating our hypothesis, also shows that we can avoid the computational cost of fine-tuning the pre-training language models and also take advantage of better explainability from the KNN results. Also, our findings reveal significant insights into the utility of informal definitions in reflecting top-level ontology concepts, highlighting the potential for developing automated approaches to help an ontology engineer during the ontology development process.

Keywords: Top-level ontology classification, Informal definition, Language model, Ontology learning

1. Introduction

In recent years, two significant areas in Artificial Intelligence (AI) have collided again. On the one hand, we have ontologies, which can be defined as *a formal and explicit specification of a shared conceptualization* (Studer et al., 1998). On the other hand, we have the advances in natural language processing (NLP) for text representation and classification with (large) language models (Devlin et al., 2018; Radford et al., 2018; Touvron et al., 2023; Jiang et al., 2024). This convergence has ushered in what can be described as a new renaissance in ontology learning, in which the aim is to generate ontologies from text

*Corresponding author. E-mail: agljunior@inf.ufrgs.br.

semi-automatically using state-of-the-art NLP techniques. While various methods have been proposed to tackle this challenge (He et al., 2022, 2023; Babaei Giglou et al., 2023), the field is still rife with open questions and opportunities for exploration.

Ontologies offer numerous advantages, particularly knowledge representation and data management. Firstly, ontologies provide a structured and standardized way to represent and organize complex information, allowing for improved data integration and interoperability (Cicconeto et al., 2022; Kulvatunyou et al., 2022; Qu et al., 2024). From this, we can retrieve information and share knowledge across different systems and resources more effectively. Secondly, ontologies enhance semantic clarity by precisely defining domain entities and their relationships, reducing ambiguity, and ensuring a common understanding of complex knowledge domains. In this context, state-of-the-art ontology engineering methodologies, such as NeOn (Suárez-Figueroa et al., 2011), recommend using a top-level ontology to ensure these two characteristics in a domain ontology.

A top-level ontology defines a foundational and high-level structure for categorizing and organizing knowledge across various domains and subject areas (Borgo et al., 2022; Guizzardi et al., 2022; Otte et al., 2022). It serves as a broad and abstract framework for representing fundamental concepts and relationships that are universally applicable and not tied to any specific field or domain. From that, top-level ontologies are the starting point for organizing and categorizing domain-specific knowledge of domain ontologies. However, classifying domain entities into top-level concepts remains a manual and time-consuming task and requires a high level of expertise in the target domain and ontology engineering. Using OBO Foundry ontologies (Jackson et al., 2021) as an example, several domain ontologies contain millions of domain entities[1]. Still, only a few thousand are aligned with a top-level ontology, i.e., have a direct or indirect relationship to a top-level ontology concept.

In this work, we addressed the problem of classifying domain entities into top-level ontology concepts using informal definitions to represent domain entities textually. From that, our main hypothesis is that the informal definitions represent semantic information that allows domain entities to be related to top-level ontology concepts. In this context, two crucial ideas were leveraged. Firstly, since top-level ontologies also contain a taxonomy of concepts, we leveraged the similarity in taxonomies, in which domain entities grouped in the same ontology concept share common features or attributes, i.e., they are more similar to each other than with other domain entities under another ontology concept. Secondly, since we are using informal definitions, we leverage the Distributional Hypothesis(Harris, 1954), which suggests that words that occur in similar contexts are likely to have related meanings, i.e., they are closer in the distributional space. Also, informal definitions provide the intended meaning of a domain entity in a particular domain (Seppälä et al., 2016) and are provided early in the ontology development process (Suárez-Figueroa et al., 2011). Thus, our goal is to show the consequences of our hypothesis for the entire structure of concepts contained in a top-level ontology. In order to evaluate our hypothesis, we proposed two supervised classification pipelines containing (1) a fine-tuned pre-trained language model for this problem. Also, we proposed a methodology for extracting the datasets to train and evaluate the classifiers and (2) a dense text vectorization with a classical machine learning classifier.

In order to validate our hypothesis, we proposed the extraction of multi-label, multi-language, and multi-resource datasets from the alignment between OntoWordNet ontology (Gangemi et al., 2003) and the BabelNet semantic network (Navigli and Velardi, 2004; Navigli et al., 2021). From this alignment, we expanded the OntoWordNet to 291 languages and various knowledge resources beyond WordNet 1.6 (Miller, 1995), such as WordNet 3.0, Wikipedia, WordNet2020, among others. In addition, we extracted

[1] https://dashboard.obofoundry.org/dashboard/analysis.html

the datasets in a multi-label format by assigning, for each domain entity, the entire branch in which it subsumes from the Dolce-Lite-Plus top-level ontology as its labels. Thus allowing the multi-inheritance representation of domain entities and also the exploration of the performance of the proposed pipelines at each level of depth of Dolce-Lite-Plus

In our experiments, we conducted three study cases. Firstly, we examined the performance of the proposed pipelines for different textual representations of domain entities. Secondly, we investigated the performance of our proposal for datasets in multiple languages. Thirdly, we evaluated the performance of our proposal by training and testing the proposed pipelines using informal definitions from different knowledge resources. As a result, we discovered that using informal definitions is the best way to represent domain entities textually to classify domain entities into top-level ontology concepts. Also, our results show that our hypothesis is valid for different languages and resources. In addition, although fine-tuning a classification model for a specific task has promising results in many fields, our work shows that employing a K-Nearest Neighbor approach using embedding representing as input has better results and avoids the computation costs of fine-tuning. Overall, our best result was achieved by using the Mistral7B language model in the pipeline (2).

The paper is organized as follows. In Section 2, we present the evolution of ontologies from philosophical concepts to indispensable tools in knowledge representation. Also, we describe the importance of precise definitions and the Distributional Hypothesis in computational linguistics. In Section 3, we review the advancements in language models from embeddings to transformer-based architectures and their impact on ontology learning. Section 4, we introduce our proposed approach of using informal definitions for classifying domain entities into top-level ontology concepts, detailing dataset extraction, classification pipelines, and training methodologies. In Section 5, we discuss the practical results and effectiveness of the proposed classification pipelines over the addressed study cases. Finally, in Section 6, we offer concluding remarks on our work.

2. Background

In this section, we discuss the significance and application of ontologies in computer science, contrasting their philosophical origins and highlighting their role in knowledge modeling. Also, we focus on top-level ontologies as general frameworks for knowledge representation across various domains. After that, we review informal definitions, highlighting the importance of clear and precise definitions for domain entities in order to disambiguate terms and align them with the expert understanding of context. Additionally, we review the Distributional Hypothesis, illustrating its foundational role in linguistics and computational linguistics, which influenced the development of word embeddings and state-of-the-art language models.

2.1. Ontologies and Top-level Ontologies

Ontologies are powerful tools to support knowledge modeling and computer science. In this context, ontology is a term originating from Philosophy that has different meanings and uses in various communities. According to Guarino et al. (2009), the most divergent senses of ontology come between Philosophy and Computer Science, wherein the first ontology meaning refers to the study of the nature and structure of things per se, independently of any further considerations, and even independently of their actual existence. On the other hand, in Computer Science, an ontology is a special kind of information object or computation artifact used to model a system's structure formally. Besides these views,

ontologies in computer science are often implemented using formal languages like the Web Ontology Language (OWL), enabling machines to process and understand structured information. From that, ontologies bridge human conceptual understanding and machine readability, thus facilitating more efficient and accurate data processing, analysis, and decision-making in complex systems and enabling semantic interoperability between different systems. Also, there are several ways to classify ontologies regarding the domain covered, the formal language used, or the level of generality of the entities defined in the ontology (Guarino, 1998; Prestes et al., 2013). In this last case, a consensus in the literature proposes four levels of generality in which an ontology can be classified: (1) top-level ontologies, (2) core ontologies, (3) domain and task ontologies, and (4) application ontologies. Based on that, in this work, we use the term "domain entity," referring to any entity described in Levels 2, 3, and 4 of these types of ontologies.

Delving deeper into top-level ontology (a.k.a, upper-level, foundational, or general ontologies) is a framework for representing knowledge across different domains, critical for knowledge modeling and ontology engineering (Suárez-Figueroa et al., 2015). This type of ontology is designed to encapsulate fundamental concepts and principles that are universally applicable, regardless of the specific domain. This includes very abstract concepts like time, space, events, objects, relationships, and qualities. From that, the primary purpose of top-level ontologies is to establish a common understanding and a universal vocabulary from which the entities in the other types of ontologies can subsume and ensure consistency and interoperability between them. In this context, several top-level ontologies are proposed, for example, BFO (Arp et al., 2015; Otte et al., 2022), DOLCE (Gangemi et al., 2002; Borgo et al., 2022), SUMO (Niles and Pease, 2001), UFO (Guizzardi et al., 2022), among others.

In this work, we focused only on DOLCE's top-level structure, particularly on the DOLCE-Lite-Plus (DLP)[2]. The DLP top-level ontology was developed based on the DOLCE foundations (Gangemi et al., 2002), extending it using other top-level concepts for describing descriptions, situations, temporal relations, information objects, actions, agents, social units, collections, and collectives. The whole taxonomic structure of DLP contains a total of 244 concepts.

2.2. *Informal Definitions*

According to Robinson (1950), a definition is conceptualized as a statement that clarifies the meaning of a term or concept. In his work, Robinson posits that definitions are intended to elucidate the essence of what is being defined, thereby distinguishing it from other entities. He emphasizes that a good definition must be clear, precise, and concise, avoiding ambiguity or circular reasoning. Definitions are not just seen as linguistic tools; they are fundamental to logical analysis and philosophical inquiry. They play an instrumental role in advancing understanding, facilitating communication, and providing a foundation for further exploration across various fields of knowledge. From this perspective, definitions are more than mere explanations of words. They are crucial elements in the structure of knowledge, aiding in the articulation and differentiation of concepts essential to intellectual discourse.

Other works, such as Seppälä et al. (2016), extend the contributions on definitions in ontologies. According to their work, the definition of the term "definition" can vary in nature, depending on the context of use and target audience. However, definitions are essential tools for communication and understanding, specifically designed to align with the cognitive and linguistic requirements of their intended context. From a linguistic perspective, definitions aim to align with a certain pre-existing lexical use, conveying the semantic value of a term by delimiting its intension and extension. They adjust the overall

[2]http://www.ontologydesignpatterns.org/ont/dlp/DLP_397.owl

lexical competence of users by enhancing their inferential competences, particularly semantic inferential competence. On the other hand, from a cognitive perspective, definitions in ontologies bring about a reconfiguration of the receiver's existing body of knowledge regarding the intended referent of the defined term. They work to augment and reconfigure the knowledge and beliefs of the user to align them more closely with those of relevant expert communities. Overall, definitions help in disambiguating terms and ensuring consistency in their use by providing clear and unambiguous intended meaning (often using Aristotelian principles) to distinguish a term from neighboring terms.

Typically, an informal definition in an ontology has a canonical form that derives from the Aristotelian form of the definition "X is a Y that Z", in which "X" is the *Definiendum*, "Y that Z" is the *Definiens*, and "is a" is the *Copula*. For example, "Rock is a material consisting of the aggregate of minerals like those making up the Earth's crust". Definiendum is the defined term, the subject of the informal definition, i.e., what the informal definition is about (e.g., "Rock"). Definiens is the part that expresses the content of the informal definition, i.e., the explanatory part of the informal definition that specifies what the Definiendum is (e.g., "material consisting of the aggregate of minerals like those making up the Earth's crust"). Copula is the linking part that expresses an equivalence or subsumption between the Definiendum and the Definiens (e.g., "is a"). In this work, we used the terms Definiendum and Definiens to refer to a specific part of an informal definition.

2.3. Distributional Hypothesis

The Distributional Hypothesis (Harris, 1954) presents a foundational idea in linguistics, suggesting that words that occur in similar contexts are likely to have related meanings. This hypothesis proposes a method for inferring the semantics of words through the analysis of their distributional patterns across texts. Harris (1954) approach to structural linguistics emphasized the importance of context in understanding linguistic meaning, proposing that the semantic attributes of words could be uncovered through a systematic examination of their usage in various linguistic environments. By focusing on the empirical analysis of language, Harris (1954) work established a framework for semantic analysis that relies on observable, quantitative data, shifting away from more introspective or purely theoretical methods. This framework has profoundly influenced the way linguists and computational linguists approach the study of language by suggesting that the meanings of words can be deduced from the patterns of their use.

This hypothesis has had a significant impact on the development of computational linguistics, particularly in creating technologies like word embeddings (Mikolov et al., 2013a,b; Pennington et al., 2014), and language models (Radford et al., 2018; Devlin et al., 2018; Touvron et al., 2023; Jiang et al., 2024). These models represent words as vectors in a high-dimensional space, where the proximity between vectors indicates semantic similarity based on their distributional properties. This approach has enabled advances in natural language processing (NLP), allowing for a more nuanced and effective machine understanding of human language. The practical applications of the Distributional Hypothesis are evident in various NLP tasks, including machine translation, information retrieval, and sentiment analysis, demonstrating its vital role in bridging linguistic theory and computational applications.

3. Related Work

In this section, we discuss the significance and application of ontologies in computer science, contrasting their philosophical origins and highlighting their role in knowledge modeling. Also, we focus on top-level ontologies as general frameworks for knowledge representation across various domains.

After that, we review informal definitions, highlighting the importance of clear and precise definitions for domain entities in order to disambiguate terms and align the domain expert's understanding of the context. Additionally, we review the Distributional Hypothesistionally, illustrating its foundational role in linguistics and computational linguistics, which influenced the development of word embeddings and state-of-the-art language models.

3.1. Language Models

The evolution of natural language processing (NLP) from word embeddings (Mikolov et al., 2013a,b) to sentence embeddings (Reimers and Gurevych, 2019) and language models (Devlin et al., 2018; Radford et al., 2018; Touvron et al., 2023; Jiang et al., 2024) marks a significant advancement in understanding and processing human language. The relation between language models and the distributional hypothesis (Harris, 1954) lies in how the models learn: by predicting words based on their contexts (for example, the words before and/or after a given word in a sentence). This predictive process inherently relies on the assumption that words with similar contexts have similar meanings as the models adjust their internal parameters to reduce prediction errors based on the contexts they observe. Consequently, language models not only learn to predict words accurately but also implicitly learn rich, context-sensitive embeddings for words and phrases that reflect their meanings and relationships, embodying the principles of the distributional hypothesis.

A significant advancement came with the development of transformer-based models, such as BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2018) and GPT (Generative Pre-trained Transformer) (Radford et al., 2018). These models revolutionized NLP by using attention mechanisms to understand the full context of a word in relation to all other words in a sentence or even across multiple sentences. In this context, BERT, in particular, marked a paradigm shift by pre-training on a large corpus of text and then fine-tuning for specific tasks, achieving unprecedented performance across a range of NLP benchmarks. Its bidirectional nature meant it could effectively understand the context from both the left and the right of each word in a sentence, providing a more thorough understanding of language. On the other hand, GPT and its successors, GPT-2, GPT-3, and GPT-4, further expanded on this concept, using massive amounts of data and computational power to generate highly context-sensitive embeddings and even generating coherent extended text.

DistilBERT (Sanh et al., 2020), RoBERTa (Liu et al., 2019), and ALBERT (Lan et al., 2019) are advancements over BERT focusing on efficiency and performance. DistilBERT reduces the model size by 40% while retaining 97% of BERT's capabilities through knowledge distillation, making it faster and lighter. RoBERTa optimizes BERT's pretraining by removing the Next Sentence Prediction objective, extending training, and dynamically changing the masking pattern, significantly outperforming BERT on major benchmarks. ALBERT introduces parameter-reduction techniques and a Sentence Order Prediction loss, achieving better results with fewer parameters and focusing on inter-sentence coherence.

T5 (Raffel et al., 2023), LLAMA (Touvron et al., 2023), and Mixtral (Jiang et al., 2024) represent significant advancements in language model architecture. T5 adopts a unified text-to-text framework, simplifying the diverse landscape of NLP tasks by treating them all as text-to-text problems, which allows for systematic study and comparison of different approaches. LLAMA represents a collection of foundation language models focusing on scalability and accessibility. Ranging from 7B to 65B parameters and trained on trillions of tokens using publicly available datasets. Mixtral focuses on fast inference with offloading techniques, making it a practical solution for using large language models on consumer hardware.

The field of language models continues to evolve rapidly, with ongoing research exploring even more sophisticated models and techniques for capturing the nuances of human language. The relation between these models and the distributional hypothesis lies in how the models learn: by predicting words based on their contexts (for example, the words before and/or after a given word in a sentence). This predictive process inherently relies on the assumption that words with similar contexts have similar meanings as the models adjust their internal parameters to reduce prediction errors based on the contexts they observe. Consequently, language models not only learn to predict words accurately but also implicitly learn rich, context-sensitive embeddings for words and phrases that reflect their meanings and relationships, embodying the principles of the distributional hypothesis.

3.2. Ontology Learning

The field of ontology learning primarily focuses on developing ontologies through automatic or semi-automatic processes (Khadir et al., 2021). This area encompasses a variety of systems, methodologies, and algorithms aimed at either fully automating the ontology creation process or facilitating certain stages of it (Wong et al., 2012; Khadir et al., 2021). In this context, with the advancements in word embeddings and language models, He et al. (2022) introduced BERTMap, an innovative system utilizing the BERT language model for ontology alignment tasks. This system performs mapping predictions and refinements, showing significant improvements over existing methods in handling large-scale ontologies. Extensive evaluations demonstrate BERTMap's better performance, especially in biomedical ontology tasks, highlighting its practicality and effectiveness in ontology alignment and knowledge integration. Although ontology alignment and ontology learning are not the same thing, they are related areas.

In Chen et al. (2023), the authors proposed a method for predicting class subsumptions within and between ontologies. The method leverages BERT, a pre-trained language model, to compute contextual embeddings of ontology classes. Custom templates are used to integrate class context and logical existential restrictions, enabling predictions of various subsumers within the same or different ontologies. Extensive evaluations with real-world ontologies show that the proposed approach significantly outperforms baseline methods, highlighting the effectiveness of the templates in handling both named class and existential restriction subsumptions. The study underscores the importance of contextual embeddings and tailored approaches for enhancing machine learning tasks in knowledge engineering.

In Babaei Giglou et al. (2023), the authors explored the application of Large Language Models (LLMs) in ontology learning. Their work investigates whether LLMs can effectively extract and structure knowledge from natural language text for ontology learning. The study conducts comprehensive evaluations using the zero-shot prompting method across nine different LLM model families, focusing on three main ontology learning tasks: term typing, taxonomy discovery, and extraction of non-taxonomic relations. The research spans various knowledge domains, including lexico semantic with WordNet Miller (1995), geographical, and medical knowledge. The findings suggest that while foundational LLMs have limitations in ontology construction, they could be effective assistants when fine-tuned, potentially alleviating the knowledge acquisition bottleneck in ontology construction.

Nowadays, only a few works have explored the intrinsic aspects of ontology learning for top-level ontologies using state-of-the-art approaches in natural language processing. Most of the works were proposed by our research group. In this context, in Lopes et al. (2022), we started the discussion about using word embedding of Definiens concatenated with the word embedding of Definiendum as input of a machine learning pipeline, which classifies the input into a restricted number of DLP concepts from OntoWordNet project (Gangemi et al., 2003). After that, in Lopes et al. (2023), we explored the performance of multiple language models of the BERT's family using as input the whole informal definition

and classifying them into all leaf concepts of DLP used in the OntoWordNet project. As another important work in this line, in Rodrigues et al. (2023), we explored the effectiveness of few-shot learning using manually generated prompts in ChatGPT for classifying domain entities from IOF (Industrial Ontology Foundry) ontology[3] into concepts of DOLCE and BFO top-level ontologies. However, ChatGPT did not correctly deal with certain finer ontological distinctions and suffered from hallucinations on several generated responses.

In state-of-the-art ontology learning approaches, it is notable that there is a lack of focus on classifying domain entities into top-level ontology concepts. While existing methodologies have made significant progress in various aspects of ontology development, like entity extraction, relation prediction and filtering, and taxonomy creation, there is a clear gap in developing approaches to the task of classifying domain-specific entities to top-level ontology concepts. This research gap highlights the need for further exploration and innovative methods to address this deficiency, which is important to enhancing the comprehensive and coherent development of well-founded domain ontologies.

4. Proposed Approach

In this section, we detailed an approach for classifying domain entities into top-level ontology concepts using informal definitions as input. In this context, we start introducing the vocabulary throughout this section. After that, we present a methodology to extract multi-label, multi-language, and multi-resource datasets by aligning OntoWordNet and BabelNet resources. Then, we advocate for using informal definitions as the optimal textual representation for domain entities, proposing two classification pipelines leveraging language models to predict top-level ontology concepts. Finally, we present a training methodology that includes data preparation techniques like the "explode" approach for dataset augmentation, aiming to effectively evaluate pipeline performance and dataset quality through strategies like k-fold cross-validation.

4.1. Technical Vocabulary

In the next sections, we used the following vocabulary:

- **Informal definition:** An informal definition is an explanation of the meaning of a term in a non-technical and understandable textual way, comprising the Definiendum and the Definiens linked by the Copula. E.g., "Rock is a material consisting of the aggregate of minerals like those making up the Earth's crust";
- **Definiendum** (plural form, **Definienda**)**:** Definiendum is the defined term, the subject of the informal definition. It's what the informal definition is about. E.g., "rock";
- **Definiens** (plural form, **Definientia**)**:** Definiens is the part that expresses the content of the informal definition. It's the explanatory part of the informal definition that specifies what the Definiendum is. E.g., "material consisting of the aggregate of minerals like those making up the Earth's crust";
- **Copula:** Copula is the linking part that expresses an equivalence or subsumption between the definiendum and the definiens. E.g., "is a";
- **Example sentence:** An example sentence is a sample phrase designed to illustrate the use of a term within a specific context. E.g., "He throws a rock at me";

[3] https://www.industrialontologies.org/

- **Top-level ontology:** A top-level ontology is a high-level, abstract, and domain-agnostic categorization framework that organizes concepts across knowledge domains. E.g., DOLCE, BFO, UFO;
- **Knowledge resource:** A knowledge resource is a repository or source of information and data. E.g., glossaries, dictionaries, ontologies, semantic networks;
- **Semantic network:** A semantic network is a graph representation of a knowledge resource where various types of relationships interconnect the domain entities. E.g., WordNet and BabelNet;
- **Taxonomy:** A taxonomy is a hierarchical classification system that organizes concepts, objects, or information into categories and subcategories based on shared characteristics.

4.2. Dataset Extraction

In this section, we describe the methodology used to extract the multi-label, multi-language, and multi-resource datasets used in this work. Firstly, we presented how we align OntoWordNet and BabelNet resources. After that, we introduce a novel algorithm for matching domain entities across these resources and the transition from a multi-class to a multi-label dataset to accommodate the complex, hierarchical relationships between domain entities and top-level ontology concepts. Finally, we present the challenges in extracting this kind of dataset, such as partial coverage, special character discrepancies, and the unbalanced nature of top-level ontologies.

4.2.1. OntoWordNet Ontology and BabelNet Semantic Network

The starting point for extracting the datasets used in this work began with the ontology from the OntoWordNet project. This ontology project established a comprehensive alignment between WordNet—a structured collection of English words grouped into sets of synonyms (synsets)—and Dolce-Lite-Plus (DLP) top-level ontology concepts. The alignment process undertaken by OntoWordNet introduced a systematic categorization among the WordNet synsets. This categorization was pivotal to the transformation of WordNet from merely a lexical database into a versatile ontology library by clearly separating concept-synsets, relation-synsets, meta-property-synsets, and individual-synsets. The output of this alignment comprises a total of 65,973 domain entities. Also, these domain entities are aligned with 120 top-level concepts from the 244 of the whole structure of Dolce-Lite-Plus.

Figure 1 provides a detailed visualization of the relationship between Dolce-Lite-Plus (DLP) top-level concepts and the corresponding WordNet synsets from the alignment provided by the OntoWordNet ontology. This figure strategically emphasizes only the direct top-level ontology concepts from DLP, under which the domain entities are subsumed. From that, the figure highlights the unbalanced nature of the OntoWordNet ontology, illustrating how certain top-level DLP concepts are associated with a large number of WordNet synsets, while others may have far fewer. This disparity is crucial for understanding the distribution and representation of concepts within the top-level ontology, offering insights into potential areas of richness or sparsity of domain entities, which will be addressed later.

In WordNet, informal definitions are divided into two parts for specific technical purposes: definiendum and definiens. The definiendum part refers to each term within a synset, a group of synonyms that share a common meaning, indicating the subject of the definiens. Conversely, the definiens provides the explanation or the descriptive content of the synsets meaning, encapsulating the essence of the concept with only one definiens assigned per synset to maintain clarity and precision. Therefore, the domain entities of the OntoWordNet ontology follow the same organization. This distinctive organization facilitates mapping the OntoWordNet domain entities to a more expansive and interconnected semantic network. One such network is BabelNet, which also integrates the comprehensive lexical coverage of WordNet extended with the extensive encyclopedic knowledge from Wikipedia. From that, aligning and

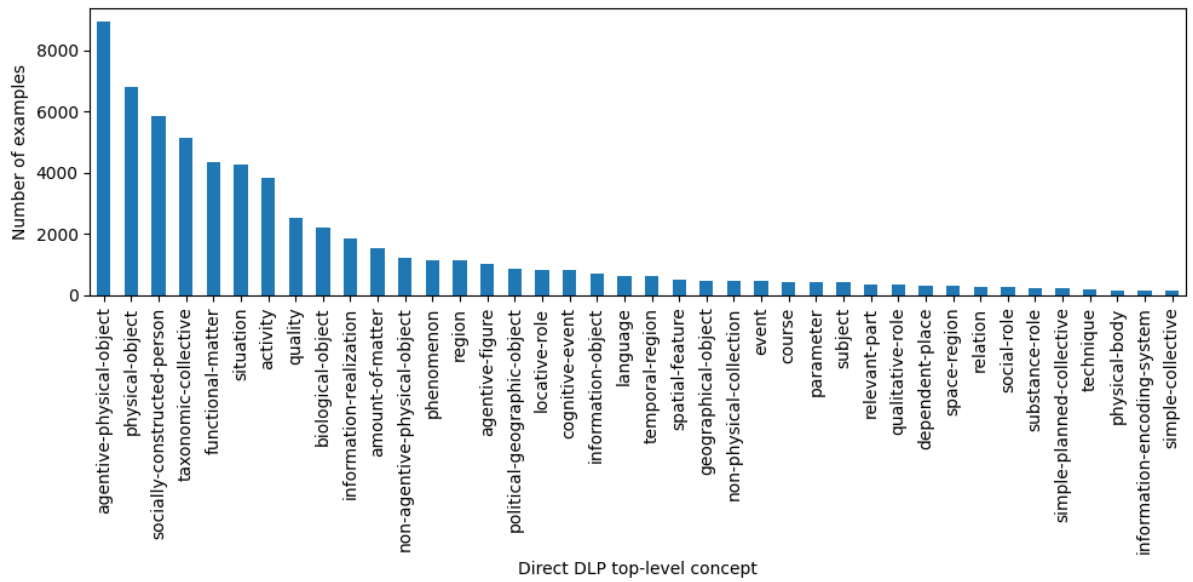

Fig. 1. Top 40 most populated direct DLP top-level concepts of WordNet synsets in the OntoWordNet.

enhancing the OntoWordNet domain entities with the multi-resource and multi-language information in BabelNet becomes possible.

4.2.2. *Matching Between OntoWordNet and BabelNet*

This work also introduces an approach to align the domain entities identified within the OntoWordNet ontology with those cataloged in the BabelNet semantic network. BabelNet encompasses more synsets than WordNet, from adding Wikipedia and synsets for more recent linguistic concepts not present in WordNet, such as streamer and influencer. This broadness of BabelNet provides an opportunity for the enhancement and expansion of OntoWordNet domain entities to multiple multi-language and multi-resource datasets. In this context, we take the advantage that both the structure of WordNet inside OntoWordNet and BabelNet share almost the same definienda to propose the Algorithm 1 to match the domain entities between them.

Algorithm 1 begins by inputs the OntoWordNet ontology OWN and the BabelNet structure BN and the initialization of the alignment *align* between OWN and BN as an empty list. After initializing *align*, we get all the domain entities from OWN, and then the algorithm loops through them. For each *domain_entity*, we get its $definienda$, $definiens$, and DOLCE-Lite-Plus (DLP) top-level concept *dlp_cls*. Once this information is extracted, we initialize the *rank* dict variable for the respective *domain_entity* and loop through all elements in $definienda$. In BabelNet, the same $definiendum$ can name multiple synsets. Based on that, we retrieved all the synsets associated with the $definiendum$ from BabelNet. In this context, since OWN is aligned with WordNet, we look only at the WordNet part inside BabelNet, reducing the number of comparisons required. After retrieving all the synsets associated with the $definiendum$, we loop through them. Inside this loop, we verify if the current *synset* is already in *rank*. If not, we initialize the *rank* dict in the *synset* position with a tuple $(0, 0)$, where the first and second elements store the similarity evaluations and the number of similarity evaluations performed, respectively. This process continues, with extracting the *Definiens bn_definiens* from the BabelNet *synset*, performing the similarity evaluation between the domain entity $definiens$ and the

BabelNet *bn_definiens* through Jaro-Winkler distance[4], and incrementing the number of comparisons. Through this ranking, we stored all BabelNet synsets' information, which matches each *definiendum* of the domain entity. After that, we loop through the counting information of all synsets in the *rank* and select the greater. Our objective with this value is to normalize the similarity sum of each synset. This is necessary to penalize synsets with a high similarity between their *Definiens* but few *Definienda* in common. Once the ranking *rank* is normalized, we select the synset with the highest similarity score as the match with the respective domain entity and add this synset to the *align* list associated with *dlp_cls*.

In addition, Algorithm 1 has an asymptotic complexity of $O(n^3)$. However, it offers a significant efficiency over a method that compares all domain entities of OntoWordNet with all entities with a complexity of $O(n^2)$. This efficiency gain is particularly relevant considering the size of BabelNet, which contains more synsets than WordNet, the basis of OntoWordNet. Since OntoWordNet comprises 86,982 domain entities, a naive approach that compares these entities with only relevant BabelNet synsets would entail up to 7.5 billion comparisons. In contrast, the proposed algorithm exploits the fact that BabelNet synsets are indexed by synset term or ID, allowing for their retrieval in $O(1)$ time (as seen in line 8 of our algorithm). Consequently, the algorithm iterates over the terms naming OntoWordNet entities and retrieves only the associated BabelNet synsets, significantly reducing the number of comparisons to a few million.

The resulting alignment contains a total of 65,018 domain entities if we consider the main dataset that comprises domain entities represented by informal WordNet definitions in English. Following the application of Algorithm 1, we transitioned from a single-language resource, relying solely on the definitions from WordNet, to the ability to harvest datasets in various languages and from a wide range of semantic resources. This advancement enabled the extraction of datasets for the task of predicting the top-level concept of a domain entity across 291 languages and from many kinds of resources, such as WordNet, Wikipedia, Wiktionary, Open Multilingual WordNet, WordNet 2020, OmegaWiki, etc. Also, this alignment made it possible to access other types of resources beyond just informal definitions, as well as example sentences of the terms, Wikipedia source page of the domain entities, images, relationships with other domain entities, etc. Integrating these diverse elements significantly enhances the breadth and depth of linguistic and semantic information within our datasets, thereby enriching the scope and utility of the datasets that can be extracted.

4.2.3. From a Multi-Class to a Multi-Label Dataset

Currently, all approaches for classifying domain entities into top-level ontology concepts consider the task in a multi-class classification scenario, i.e., a domain entity is assigned to one and only one label. However, this scenario has several drawbacks if the motivation is to help the decision-making process of an ontology engineer during ontology development. Firstly, it is expected a significant class imbalance (as presented in Figure 1) due to the nature of the concepts in top-level ontologies, which can negatively affect the classification accuracy for less populated classes. Secondly, a more general top-level concept for a domain entity with high accuracy can help better decision-making than a specific top-level concept with low accuracy. In this context, we aimed to transform the multi-class dataset resulting from OntoWordNet and BabelNet alignment into a multi-label dataset, where a domain entity is assigned to the entire branch of the top-level ontology in which it subsumes.

The first step in this transformation process involves analyzing the top-level ontology hierarchy for the top-level concept of each domain entity in the dataset. In this analysis, if a domain entity belongs to a

[4]Jaro-Winkler distance fits our purposes in this work perfectly because the compared sentences are almost identical if the compared entities are the same, and the algorithm is remarkably optimized for such computation.

Algorithm 1 The algorithm used to align OntoWordNet and BabelNet

Input: OntoWordNet OWN; BabelNet BN
Output: Alignment *align* between OntoWordNet and BabelNet

Initialize *align* as an empty list.
$domain\_entities \leftarrow DomainEntities(OWN)$
for *domain_entity* in *domain_entities* **do**
  $definienda \leftarrow Definienda(domain\_entity)$
  $definiens \leftarrow Definiens(domain\_entity)$
  $dlp\_cls \leftarrow Concept(domain\_entity)$
  Initialize *rank* as an empty dict.
  for *definiendum* in *definienda* **do**
    $synsets \leftarrow BabelNetSynsets(definiendum, 'WN', 'EN')$
    for *synset* in *synsets* **do**
      if $synset \notin rank$ **then**
        Initialize $rank[synset]$ as a tuple $(0, 0)$
      end if
      $bn\_definiens \leftarrow Definiens(synset)$
      $rank[synset][0] \mathrel{+}= JaroWinklerDistance(definiens, bn\_definiens)$
      $rank[synset][1] \mathrel{+}= 1$
    end for
  end for
  $max\_count \leftarrow \max(rank[synset][1]\ for\ synset\ in\ rank)$
  for *synset* in *rank* **do**
    $rank[synset][0] \leftarrow rank[synset][0]/max\_count$
  end for
  $best\_synset \leftarrow \arg\max(rank[synset][0]\ for\ synset\ in\ rank)$
  Add the tuple (*best*, *dlp_cls*) to *align*
end for
return *align*

child top-level concept, it also inherently belongs to its parent top-level concepts. For instance, consider the domain entity "rock." In a multi-class dataset, "rock" might be classified under "Amount of Matter." This classification is based on the entity's characteristics that match the "Amount of Matter" concept. If "Amount of Matter" is categorized as "Endurant" in the Dolce-Lite-Plus top-level ontology, then "rock" also qualifies as "Endurant" in a multi-label dataset context.

Considering the problem of classifying domain entities into top-level ontology concepts as a multi-label classification task also enables the representation of the multiple inheritance characteristic of the domain entities. Multiple inheritance occurs when a domain entity inherently belongs to more than one parent concept within the top-level ontology hierarchy. For example, suppose a domain entity like "river" fits into both "Physical Object" and "Geographical Feature" top-level concepts due to its dual characteristics. In that case, it showcases the need for a multi-label dataset over a restrictive multi-class dataset since, in the multi-class dataset, we have two examples with the same features but different labels, which downgrade the accuracy of a classification model. From that, in the multi-label dataset, we merged the labels of both examples into a single example. Thus, from the 65,018 examples in the

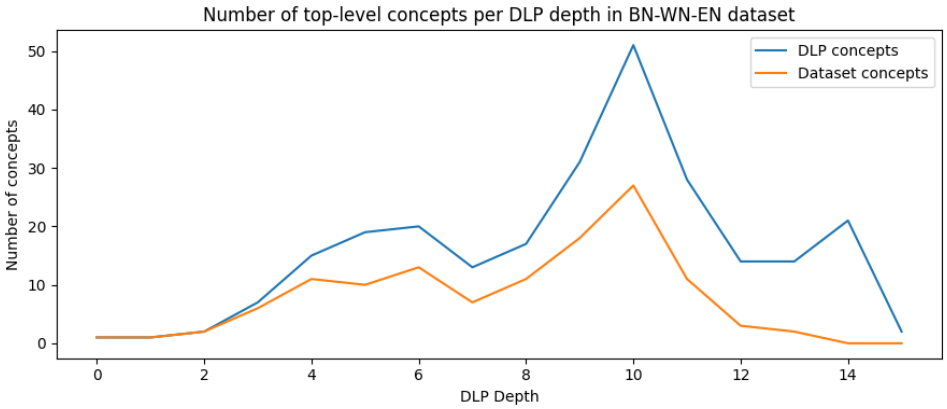

Fig. 2. The number of top-level concepts per level in Dolce-Lite-Plus and from the alignment between OntoWordNet and Babel.

original dataset, we reduced to 61,483 after considering this multiple inheritance processing.

When we merge the labels of examples with multiple inheritance, a new problem arises: disjoint labels. The issue of disjoint labels in a multi-label dataset occurs when two or more labels that should not co-occur due to their inherent conceptual or logical separation are assigned to the same domain entity. This problem poses a significant challenge in classifying domain entities into top-level ontology concepts, where the hierarchical and logical structure of the ontology must be preserved. For instance, suppose a domain entity "justice" is erroneously assigned to both "Abstract" and "Physical Object" labels. This creates a contradiction, as these top-level concepts are fundamentally disjoint within the Dolce-Lite-Plus. Such misclassification disrupts the integrity of the dataset and the top-level ontology, leading to inaccuracies in a classification model. In this context, we removed from the multi-label dataset all examples in which any of their labels are disjoint with each other. Thus, we reduced the number of examples from 61,483 after the multiple inheritance processing to 59,666 after applying the filter to remove disjoint labels, and this is the final number of dataset instances used in our experiments.

4.2.4. Coverage and Limitations

The OntoWordNet project mapped WordNet synsets to 120 of the 244 top-level concepts in Dolce-Lite-Plus (DLP). This means that OntoWordNet covers less than half of the DLP top-level concepts. However, it is essential to consider that DLP's structure is hierarchical. Thus, if OntoWordNet does not include a specific top-level concept, it must still include the broader concept that encompasses it. Figure 2 compares level by level the number of top-level concepts presented in the DLP and the number of top-level concepts in the OntoWordNet. As presented, after level 2 of the DLP hierarchy, there is a discrepancy between the number of top-level concepts between DLP and OntoWordNet, which mostly follows the same growth pattern as DLP until level 12. Based on this, we can say that although some top-level concepts are not present in OntoWordNet, more generic concepts that it subsumes will still be represented in OntoWordNet.

As another limitation, OntoWordNet does not handle special characters in the definiendum of WordNet synsets, i.e., the special characters are filtered out before the alignment with DLP. On the other hand, BabelNet indexes its synsets with special characters such as underscores, percent signs, or non-ASCII characters. Based on that, these discrepancies can cause searches to fail or return incorrect data if not adequately handled. In this context, considering that we have a BabelNet synset that was erroneously

retrieved, the ranking strategy adopted in Algorithm 1 can handle this problem by assigning a lower similarity score between the two compared definiens because they are different. Also, the ranking approach further reduces the value of the erroneous retrieved synset because the compared synsets share fewer definiendum than the right synset. In the other case, if there are no results in the BabelNet search for all definienda of an OntoWordNet domain entity, we do not include this domain entity in the alignment between OntoWordNet and BabelNet. This resulted in a decrease in the number of domain entities from 65,973 examples of original OntoWordNet to 65,018 of the alignment.

As a final limitation, top-level ontologies often exhibit an unbalanced nature due to the inherent complexity and diversity of the concepts they aim to represent. The unbalanced nature can be perceived as the difference in the number and the depth of different branches in the top-level ontology hierarchy. Also, some top-level concepts can represent only a limited number of domain entities due to their restrictions, and others can represent a broader number of domain entities because they are not so restrictive. Also, this imbalance can be attributed to the differential emphasis placed on certain areas or categories over others, reflecting the subjective priorities of the ontology designers or the specific needs of the intended application domain. Consequently, datasets derived from these top-level ontologies tend to inherit this imbalance, manifesting in skewed distributions of instances across different categories. Based on that, the performance of classification models that use such datasets can be negatively affected, and approaches to handle data augmentation in this scenario tend to be complex.

4.3. Textual Representation of Domain Entities

In this work, we aimed to propose an approach to classify domain entities into top-level ontology concepts using the textual representation of these domain entities. In this context, we hypothesize that informal definitions are the best way to represent domain entities textually, and informal definitions represent semantic information that allows domain entities to be related to top-level ontology concepts. These hypotheses leveraged two crucial ideas: similarity in taxonomies and distributional hypothesis over informal definitions.

A taxonomy is a classification system that organizes concepts, entities, or information into categories based on common characteristics or criteria. Also, taxonomies are the basis for ontologies and top-level ontologies, facilitating a more nuanced and hierarchical organization of knowledge. Based on that, domain entities grouped in the same top-level ontology concept share common features or attributes, i.e., they are more similar to each other than with other domain entities under another top-level concept. For example, all domain entities grouped under the "Amount of Matter" top-level concept, such as "gold", "iron", and "silver", have a greater similarity score because they share common characteristics about physical substances or materials. These entities are inherently more akin to one another when contrasted with entities classified under a completely different concept, such as "evening", "night", and "day" under the "Temporal Region" top-level concept.

Informal definitions provide the intended meaning of a domain entity in a particular context through quick and easy-understanding textual explanations. Usually, these informal definitions are written using the Aristotelian format ("X is a Y that Z"), containing the definiendum "X," the copula "is a" and the definiens "Y that Z." As discussed before, the definiendum is the defined term, the copula is the hierarchical relationship between the definiendum and the definiens, and the definiens is the explanatory part of the informal definition. In this context, the definiens generally contains related terms to the definiendum to define its meaning, i.e., it is expected that the definiens includes the information necessary to guarantee the intended meaning of the definiendum in a particular context. Table 1 describes examples

Table 1
Example of domain entities and their informal definitions extracted from BabelNet.

Definiendum	Informal Definition
Gold	Gold is a soft yellow malleable ductile (trivalent and univalent) metallic element; occurs mainly as nuggets in rocks and alluvial deposits; does not react with most chemicals but is attacked by chlorine and aqua regia.
Iron	Iron is a heavy ductile magnetic metallic element; is silver-white in pure form but readily rusts; used in construction and tools and armament; plays a role in the transport of oxygen by the blood.
Silver	Silver is a soft white precious univalent metallic element having the highest electrical and thermal conductivity of any metal; occurs in argentite and in free form; used in coins and jewelry and tableware and photography.
Day	Day is the time after sunrise and before sunset while it is light outside.
Night	Night is the time after sunset and before sunrise while it is dark outside.
Evening	Evening is the latter part of the day (the period of decreasing daylight from late afternoon until nightfall).

of informal definitions from the dataset described in Section 4.2 for the domain entities "gold", "iron", "silver", "evening", "night", and "day". As we can see, the informal definitions contain unique characteristics to guarantee the meaning of each domain entity. However, the informal definitions also contain common characteristics that allow us to group different domain entities. From that, we can create a group with "gold", "iron", and "silver", and another group with "evening", "night", and "day". Although we can easily create these groups using just our feelings, certain aspects guide our decision process. For example, the informal definitions of the first group contain terms related to matter and its composition, reactions, and practical usages. As for the second group, the informal definitions contain terms related to time, period of the day, and certain aspects of the temporal order in which they occur.

Since the informal definitions of similar domain entities contain related content, we can take advantage of the distributional hypothesis, which says that words that occur in similar contexts are likely to have related meanings. The assumption is that in distributional space, the domain entities "gold", "iron", and "silver", and "evening", "night", and "day" are closer to each other in their group, based exclusively on their informal definitions. Based on that, we can use state-of-the-art language models, which ensure the distributional hypothesis, in order to encode the informal definitions in an embedded form. Figure 3 describes an example of the distribution of the domain entities using the embedding of their informal definition with the BERT-Base language model. In this figure, we employed Principal Component Analysis (PCA) to reduce the BERT dimension size to 2D for better visualization. As presented, our assumption is valid for this example, i.e., based on the embedding from BERT, we obtained the same groups as the ones we supposed based on our feelings and based on a more detailed analysis of what each group contains in its informal definitions.

From the combination of the discussed topics (similarity in taxonomies and distributional hypothesis over informal definitions), we obtained our original hypothesis, which says that informal definitions represent semantic information that allows domain entities to be related to top-level ontology concepts. Our hypothesis follows the following assumptions: (1) top-level ontologies group domain entities that have common attributes or properties; (2) informal definitions are textual representations of domain entities that encapsulate some attributes or properties that express the intended meaning in a particular context; (3) based on the distributional hypothesis, we can group similar domain entities based on the embedding representation of their informal definitions.

So far, we have presented our proposal to use informal definitions to represent domain entities. However, the datasets extracted through the alignment of OntoWordNet and BabelNet (as given in Section 4.2) present other ways of textually representing domain entities, such as using the definiendum, definiens, and an example sentence. As we discussed, the definiendum and the definiens are the parts of informal definitions. Definiendum (or term) is the shortest way to represent a domain entity since the definiendum names a domain entity through a combination of one or more words. Although definienda with similar meanings are closer because of the distributional hypothesis, they can be polysemous, i.e., a single definiendum can have multiple meanings. Using the examples in Table 1, the definienda "gold" and "silver" could also represent domain entities about color. However, colors are generally considered qualities in top-level ontologies rather than the amount of matter, like in the examples. From that, polysemy is certainly a problem that would cause unwanted effects on an approach to classifying domain entities into top-level ontology concepts. On the other hand, although the definiens is the explanatory part of an informal definition and it is expected that the definiens does not have problems with polysemy, a knowledge resource (e.g., WordNet) should have only one definiens per domain entity to ensure its precision. This characteristic results in a reduced dataset size if we consider only the definiens of a domain entity, which could be increased by combining this single definiens with multiple definienda, as in the case of informal definitions.

A good candidate against informal definitions is using example sentences from where the domain entities appear (in the form of definiendum). In this context, example sentences accurately representing domain entities pose significant challenges. The first challenge regards the polysemy problem of the target definiendum in the example sentence. In this situation, if the example sentence is not previously curated, its extraction requires using a word sense disambiguation technique to ensure the right sense of the target definiendum. Another challenge regards the length of an example sentence. Longer sentences can provide more context, aiding in comprehending complex or nuanced entities. However, these demand models with larger sequence lengths or context window sizes may surpass the processing capabilities of certain embedding models. On the other hand, while computationally more feasible, shorter sentences often lack the necessary detail to convey the full usage of a domain entity. As a final challenge, the target definiendum in the example sentence might not always be clear since any definiendum in the example sentence can be the target, and the same embedding represents all of them. The latter chal-

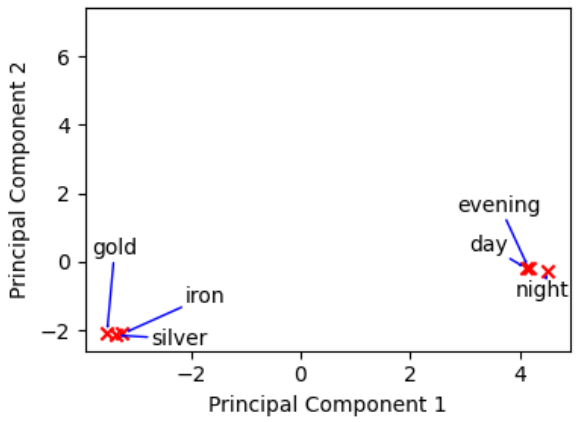

Fig. 3. Distribution of domain entities based on the embeddings of their informal definitions using BERT-Base language model.

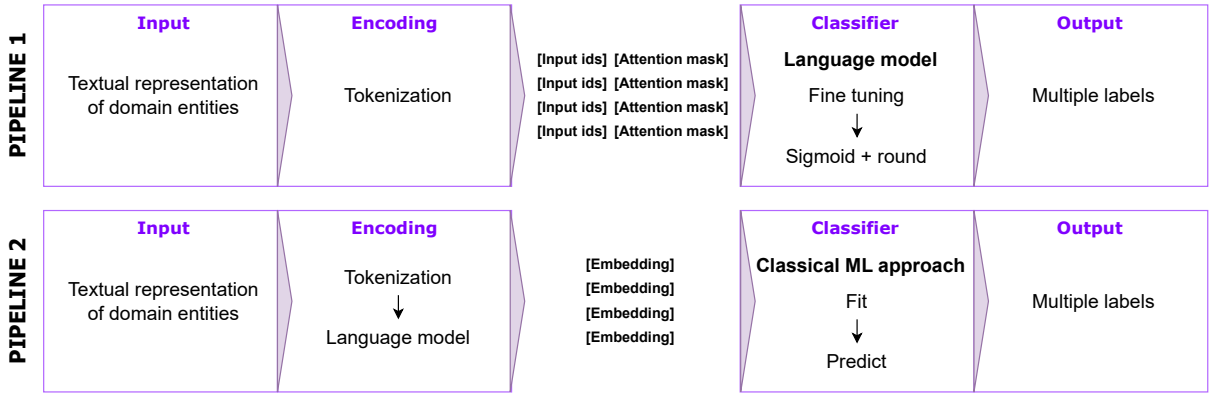

Fig. 4. Proposed pipelines for classifying domain entities into top-level ontology concepts using the textual representation of these domain entities.

lenge can be mitigated by combining the target definiendum with the example sentence, like in informal definitions that combine the definiendum with the definiens.

Based on the approaches for representing domain entities textually that we access from our alignment between OntoWordNet and BabelNet, we assume that the informal definition is the best option for several reasons: it is curated by a domain expert or can be retrieved from existing knowledge sources (like WordNet and Wikipedia); it is provided early in the ontology development process; its size is relatively small with an average of few dozen of words, which requires smaller sequence length and smaller context window from embedding models, then reducing computational costs; it does not suffer from polysemy problems since different domain entities always have different informal definitions. In contrast, the most significant disadvantage of informal definitions is that we depend on their precision, i.e., the informal definition needs to express sufficient characteristics to ensure the meaning of a domain entity. In this context, this precision can be complex to guarantee in some cases, such as for a domain entity that is not well understood yet.

4.4. Classification Pipelines

In this work, we aimed to use the distributional hypothesis over the informal definitions in order to classify domain entities into top-level ontology concepts. Also, the distributional hypothesis is the basis for state-of-the-art (large) language models (LLMs) that are trained on a vast amount of textual data. In this context, we propose the use of LLMs in two different pipelines: (1) fine-tuning the LLM weights to our specific task; (2) using the output embedding from the LLM as input of a classical machine learning approach. Figure 4 describes each step of each classification pipeline. The primary distinctions between these pipelines lie in their Encoding and Classifier steps, while they share the same Input and Output information.

Pipelines 1 and 2 in Figure 4 use the textual representation of the domain entities as input. As discussed in Section 4.3, we can access several textual representations of the domain entities from our alignment between OntoWordNet and BabelNet. Based on that, we can consider the informal definition, definiendum, definiens, example sentence, and the combination of definiendum and example sentence as possible candidates in the Input step. Also, in both pipelines, we consciously choose not to apply traditional text preprocessing techniques, such as lowercasing, stop word removal, and lemmatization, prior to inputting the text into the Encoding step. This decision is rooted in the desire to preserve the original

content and structural integrity of the textual data, which we believe plays a critical role in understanding the nuanced meanings and contextual cues inherent in language.

In the Encoding step, both pipelines use a pre-trained tokenizer to break the input text into smaller units (tokens). This process is fundamental to both pipelines, transforming unstructured text into a structured form that an LLM can interpret. In both pipelines, the outputs of the tokenizer are the Input IDs and the Attention Mask from the input text. In Pipeline 1, these tokenized forms are directly fed into a fine-tuned LLM. However, for Pipeline 2, we introduce an additional transformation step, converting tokenized text into embedding arrays. Embeddings are dense, high-dimensional vectors that capture the semantic essence of tokens. Also, embeddings provide a continuous, compact representation that encapsulates word meanings and their relationships in a vector space. This property makes embeddings particularly advantageous for classical machine learning models, which use numerical input to find underlying patterns in the data.

In the Classifier step, for Pipeline 1, we used a pre-trained LLM. During the training stage, we aimed to fine-tune this LLM for our specific task of classifying domain entities into top-level ontology concepts. During the classification stage, we used a sigmoid activation function to handle a multi-label classification scenario, i.e., an input can be assigned to multiple labels. The sigmoid function is particularly well-suited for this purpose because it converts the output of the LLM for each label into a probability score between 0 and 1, independently of each other. After that, we rounded the scores in order to get the predicted labels. As for Pipeline 2, we used a classical machine-learning approach that can input embedding and output multiple labels, such as K-Nearest Neighbor (KNN), Decision Tree, Random Forest, adaptations of Support Vector Machine for multi-label classification, ensemble models, etc. In our view, classical machine learning algorithms are designed to recognize patterns and make decisions based on numerical features. For instance, similar informal definitions are positioned closer in the embedding space, allowing a KNN model to infer the top-level concept of a domain entity based on its top-k closest neighbors.

Although the described pipelines follow the same architecture as most proposals for text classification with some modifications in the kind of input and output, we used them to validate our hypothesis (described throughout this section). The principle behind Pipeline 1 is that since LLMs have been trained on diverse and extensive textual datasets, they inherently understand the context and semantic relationships between words. Also, the fine-tuning process further adapts the LLM to recognize the specific patterns to make it more effective for our classification task. In Pipeline 2, the principle is that the distributional hypothesis from the informal definitions embedding also approximates domain entities of the same top-level ontology concept (as presented in Section 4.3). In this case, the computationally expensive process of fine-tuning can be avoided, and classical machine-learning approaches can be employed to ensure better transparency and interpretability in the classification process.

4.5. Training Methodology

During this entire section, we talked about dataset extraction and the relation between informal definition, distributional hypothesis, and top-level ontology concepts. From the dataset perspective, each domain entity is represented by a row containing three columns: the definienda, the definiens, and the labels. Since each knowledge resource, such as WordNet or Wikipedia, only includes one definiens per domain entity and a domain entity can have more than one definiendum, we can use an approach to increase the dataset size by breaking the elements in the definienda column into multiple rows, where each row contain only one definiendum in the definienda column, and the values of the other columns

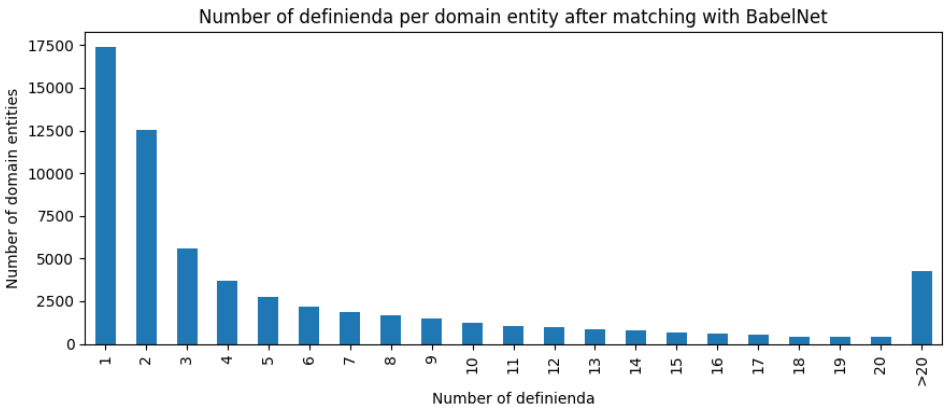

Fig. 5. The number of definienda of the domain entities in the dataset.

are replicated. In data preparation and augmentation, this technique is called "explode". For example, the "gold" domain entity in Table 1 has three different definienda in BabelNet: gold, Au, and atomic number 79. From that, we create a new row in the dataset for these three definienda and keep the same definiens and labels. Figure 5 presents the number of definienda of the domain entities in the dataset from the alignment between OntoWordNet and BabelNet.

Considering that most of the dataset domain entities have more than one definiendum (as presented in Figure 5), the "explode" approach helps increase the number of examples without artificially generating data. For instance, we can expand the dataset from 59,666 to 387,814 examples. However, this choice has implications for the performance of a classification model. Considering the example of the "gold" domain entity (from Table 1), we followed the following sequence of decisions: used Pipeline 2 (described in the previous section) with a KNN machine-learning model with $k = 3$; trained this model with two of the three new rows generated from "gold" domain entity; and predict the labels of the last remaining "gold" row. Based on the distributional hypothesis, the three closest examples predicted by KNN also include the two variations of the "gold" domain entity because they share almost the same informal definition with just a change in the definiendum part. This fact also occurs in Pipeline 1 but is more easily explainable using a KNN approach.

The cases of having an example of the domain entity with the same definiens but different definienda in the dataset and not having any example of the domain entity in the dataset are both valid. However, we must consider both for evaluating approaches for classifying domain entities into top-level concepts using informal definitions. In this context, a common approach used to evaluate the performance of a classification model and the quality of the datasets is the k-fold cross-validation, which splits the input dataset into k different fold combinations where $k - 1$ folds are used to train the model, and one fold is used to validate the model. From that, we can use the "explode" approach before or after the k-fold split. If we choose to "explode" before the k-fold split, we intend to evaluate the performance of the classification model for new informal definitions of the same domain entity. On the other hand, if we choose to "explode" after the k-fold split, we intend to evaluate the performance of the classification model for new domain entities.

5. Experiments

In this section, we present three study cases to validate our hypothesis that the informal definitions represent semantic information that allows domain entities to be related to top-level ontology concepts. Each study case addresses the same task: classifying domain entities into top-level ontology concepts. However, each one has different objectives. In the first study case, we performed several experiments to evaluate the proposed pipelines' performance using different text representation approaches for domain entities. In the second study case, we validated our hypothesis for other languages. In the third and final study case, we conducted experiments with several different language models in the proposed pipelines in order to validate our hypothesis in a cross-resource scenario.

In addressing this challenging task of classifying domain entities into top-level ontology concepts with unbalanced labels, this study used a stratified k-fold cross-validation approach with $k = 10$. The methodology involves partitioning the dataset into k equally sized folds, ensuring each fold maintains the same proportion of class labels observed in the original dataset. For each cross-validation iteration, $k-1$ of these folds are amalgamated to form the training dataset upon which the model is trained. The remaining single fold is the test dataset for evaluating the classification model's performance. This process is systematically repeated k times, with each fold getting the opportunity to be the test dataset exactly once. This process not only allows for a more robust assessment of classification model performance by mitigating the bias towards the majority class but also enhances the generalizability of the classification model across different subsets of the data. Also, we explored the variations that the "explore" approach for data augmentation (discussed in Section 4.5) causes in the results of the study cases.

In order to accurately analyze the results, we used the macro F1-score (Equation 1) to evenly assess classification performance across all labels in datasets with imbalanced labels. Also, we conducted the experiments on a machine equipped with an Intel i7-10700 CPU (4.8GHz), 32 GB of RAM, and a GeForce RTX 3060 GPU with 12GB of VRAM[5].

$$F_{\text{macro}} = \frac{1}{N}\sum_{i=1}^{N} 2 \cdot \frac{\text{precision}_i \cdot \text{recall}_i}{\text{precision}_i + \text{recall}_i} \tag{1}$$

where F_{macro} is the macro F-score; N is the number of labels; i is the ith class in N; precision_i is the precision for the ith class, defined as the number of true positive predictions divided by the total number of positive predictions (true positives plus false positives); recall_i is the recall for the ith class, defined as the number of true positive predictions divided by the total number of actual positives (true positives plus false negatives).

5.1. Study case 1: Textual representation of domain entities

In this study case, we performed a series of experiments about the effectiveness of several textual representations for domain entities to classify domain entities into top-level ontology concepts. Also, in the experiments, we compared the two pipelines proposed in Section 4.4. For Pipeline 1 (as presented in Figure 4), we employed fine-tuning the BERT-Base language model for our specific task. On the other hand, for Pipeline 2, we also used the BERT-Base language model to generate the embedding

[5]The source code and data are available at <ommited during reviewing process>

Table 2
The number of examples in each text representation dataset.

Dataset	Original size	After preprocessing	After "explode"
Informal definition	65,018	59,666	387,814
Definiendum	65,018	59,666	387,814
Definiens	65,018	59,666	59,666
Example sentence	8,417	7,329	7,329
Definiendum + Exemple sentence	8,417	7,329	43,101

representation of the input text and fed a K-Nearest Neighbors (KNN) classifier to predict the output labels.

Table 2 presents a comparative view of dataset sizes of each textual representation approach at various preprocessing stages. For the informal definition and definiendum datasets, the original size is 65,018, which was reduced to 59,666 after merging domain entities with multi-inheritance and removing each one that presents disjoint labels (as discussed in Section 4.2.3). After using the "explode" approach for data augmentation, the size expands dramatically to 387,814. In contrast, the definiens dataset remains unchanged at 59,666 examples because each domain entity has only one definiens. The example sentence dataset starts at a much smaller original size of 8,417 and decreases to 7,329 after preprocessing, with no increase after the "explode" step. The combined definiendum plus example sentence dataset mirrors this pattern, starting at 8,417, reducing to 7,329 after preprocessing, and increasing to 43,101 after the "explode" process.

Figure 6 presents the macro F-score results achieved by using the informal definitions as input for Pipelines 1 and 2. In this figure, the specific results for using the "explode" approach before and after the k-fold split are presented. Notably, when we used "explode" before the k-fold split, the results were more expressive than when we used "explode" after, with more than 90% of average macro F-score. Also, both pipelines present a stable performance until the 14th level of the Dolce-Lite-Plus (DLP) taxonomy. Overall, Pipeline 1 showed better results in relation to Pipeline 2, specifically in the experiment using "explode" after the k-fold split.

Figure 7 describes the macro F-score result achieved by combining definiendum and example sentences as input for the evaluated classification pipelines. The figure shows that both pipelines experience

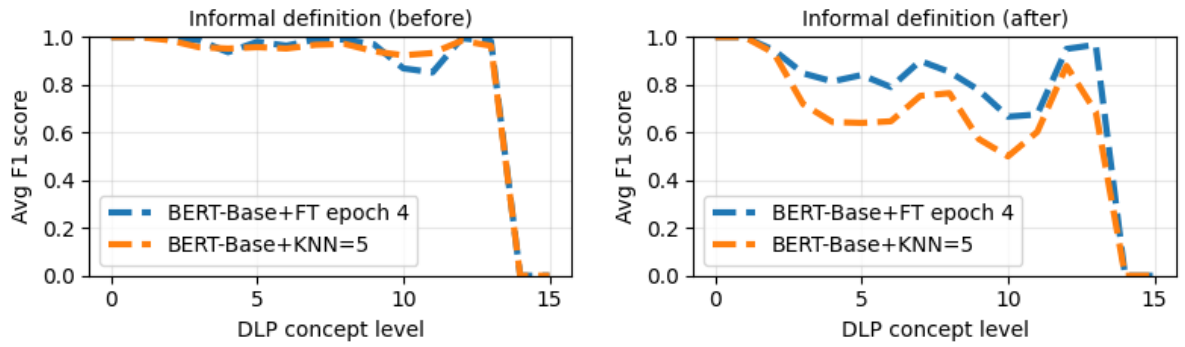

Fig. 6. Average macro F-score for each DLP level using informal definitions as input of the pipelines. The figures present the pipeline's results using the "explode" approach before and after the k-fold split, respectively.

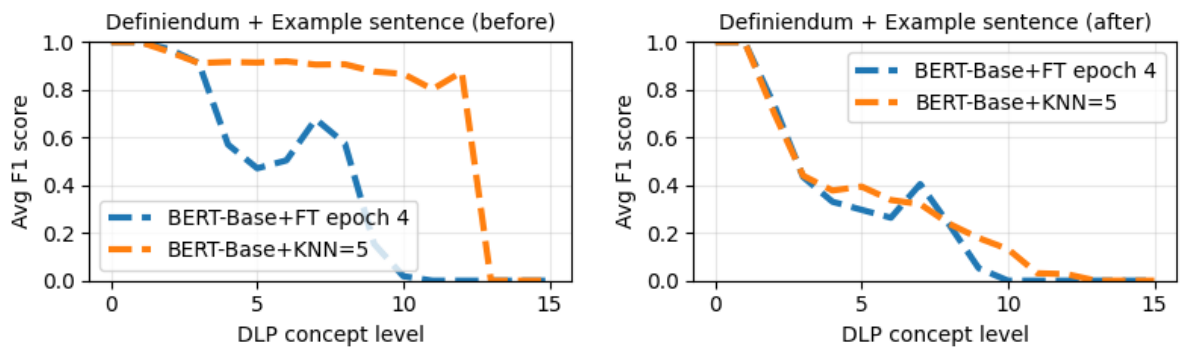

Fig. 7. Average macro F-score for each DLP level using the combination of definiendum and example sentence as input of the pipelines. The figures present the pipeline's results using the "explode" approach before and after the k-fold split, respectively.

a decline in performance as the concept level increases, with the most significant drop occurring after the 10th level, suggesting that the size of datasets can impact the results or this text representation approach is worse than informal definitions. This performance trend is consistent across the two experiments, with the "explode" process applied before and the other where the "explode" is applied after the k-fold split. Despite the performance reduction at higher levels, the BERT-Base model with the KNN approach (Pipeline 2) consistently outperforms the fine-tuned BERT-Base model (Pipeline 1), suggesting that Pipeline 2 confers a more robust understanding of the textual nuances related to domain entities, mainly using "explode" before.

Figure 8 shows the experiments using only the definiens and the example sentences of the domain entities as input of the evaluated pipelines. For the example sentences, the performance decreases progressively as the levels increase, indicating a potential challenge in capturing the full semantic scope required for classification at higher levels of specificity within the taxonomy. Using only definiens, there is a notable variation in the macro F1 score, achieving higher scores than using only example sentences in all DLP levels. The result suggests that since definiens container the descriptive part of informal definitions, this kind of text representation approach is better than using only example sentences. Also, the results show that Pipeline 2 obtained a slightly better macro F-score than Pipeline 1.

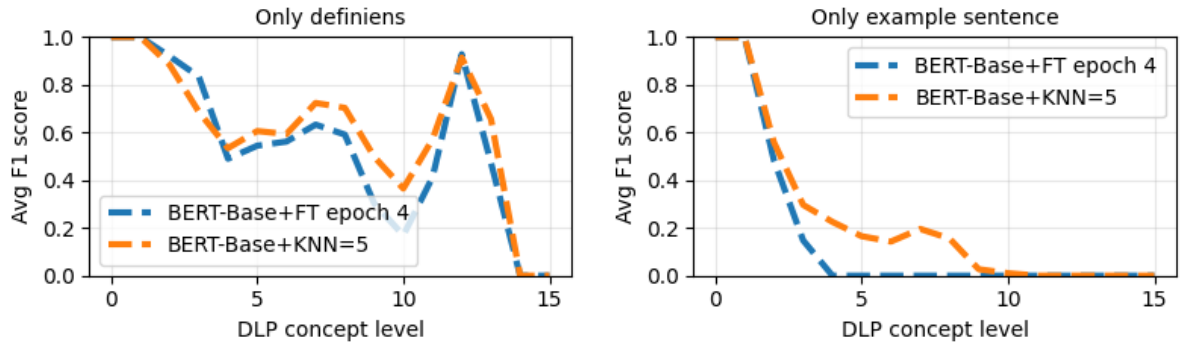

Fig. 8. Average macro F-score for each DLP level using the definiens (left) and the example sentences (right) as inputs of the evaluated pipelines.

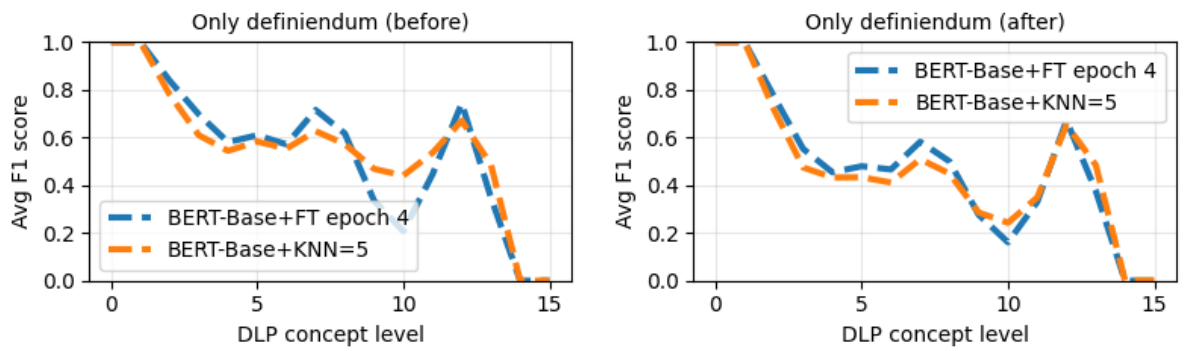

Fig. 9. Average macro F-score for each DLP level using definiendum as input of the pipelines. The figures present the pipeline's results using the "explode" approach before and after the k-fold split, respectively.

Figure 9 presents the experiments using the definiendum as input of the evaluated pipelines. The results show that the average macro F-score decreases as the DLP concept level increases. However, in both cases of using the "explode" approach before or after, there is a notable increase in F1 score at certain levels, for example, in the 6th and 12th levels. Although using only the definiendum for textually representing domain entities has the drawback of polysemy, the results were similar to those using only the definens, a form of representation that does not suffer from polysemy. Overall, both pipelines obtained similar average macro F-score performances in this experiment.

In this study case, we evaluated the performance of several textual representations of domain entities to classify domain entities into top-level ontology concepts. In this context, although several approaches achieved promising results, such as using only the definiens or only the definiendum or using the combination of definiendum and example sentences, the informal definitions proved to be the best way to represent domain entities in this task. Among the advantages is the possibility of extracting larger datasets, not suffering from the problem of polysemy, and presenting a better average macro F-score for both training strategies (before and after) from using the "explode" approach for data augmentation. In addition, Pipeline 2, which adopted the contextual embeddings from the BERT-Base language model and used a KNN classifier to classify them into top-level ontology concepts, achieved the average better stability and performance across all the experiments conducted in this study case.

5.2. Study case 2: Multi-language experiments

In this study case, we used the results of the previous study case that prove that informal definitions are the best way to represent domain entities and that Pipeline 2 achieved better stability and performance of the macro F-score in classifying domain entities into top-level ontology concepts. From that, in this case, we evaluate our hypothesis across informal definitions from different languages. In this context, Table 3 presents the number of examples in each different language dataset across the same preprocessing and data augmentation approaches carried out in the previous study case for informal definitions. In detail, the English dataset presents 2 or 3 times more examples than most of the other language datasets. Although this difference affects classification performance, we will evaluate the results using the "explode" approach before and after the k-fold split. Also, the training and test samples are in the same language for every experiment.

Table 3
The number of examples in each different language dataset.

Dataset	Original size	After preprocessing	After "explode"
English	65,018	59,666	387,814
Spanish	25,742	23,209	192360
German	25,438	23,209	149,446
Persian	20,361	18,596	141,817
French	26,173	23,967	141,730
Russian	23,724	21,634	137,830
Arabic	21,243	19,446	129,790
Chinese	20,776	18,937	125,147
Swedish	23,290	21,339	120,950
Japanese	21,229	19,876	115,496
Dutch	22,561	20,642	108,541
Portuguese	21,826	20,001	106,785
Italian	22,279	20,332	103,778

Figure 10 presents the macro F-score results for the experiment using datasets from different languages. In this experiment, we used Pipeline 2 with the BERT-Base language model and a KNN classifier to evaluate our hypothesis for every language dataset in Table 3. Based on the results, using the "exploding" approach before the k-fold split, we achieved a higher macro F-score across all evaluated languages. Spanish and English lead with better scores, followed closely by Persian, and while Japanese and Russian trail at the lower end, the scores are still high. However, using "explode" after the k-fold split, all the macro F-scores decreased, with English standing out as the highest scorer, followed by French, Spanish, and Portuguese. This result suggests there are challenges in embedding quality, issues related to the language's complexity, or the dataset's size matters for this task using the "explode" approach after the k-fold split. Overall, the scores are relatively close, reflecting consistent performance but with a noticeable advantage for languages with potentially richer informal definitions resources or more straightforward syntax for the task.

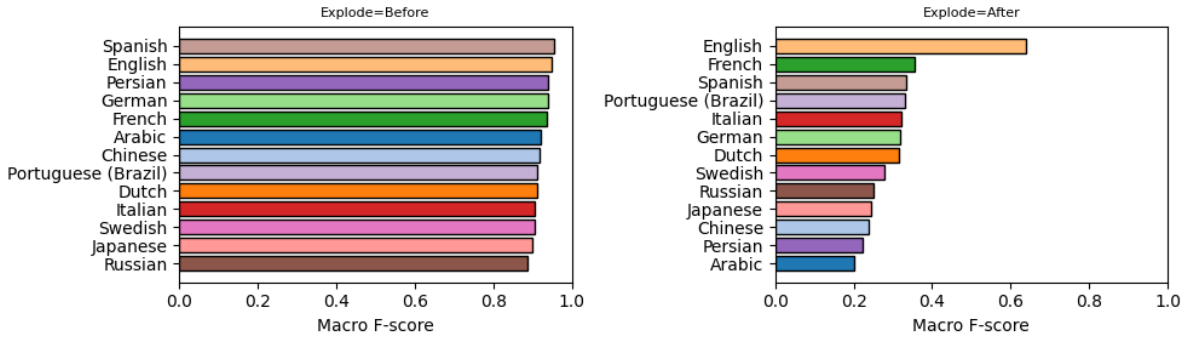

Fig. 10. Macro F-score results for each language dataset using Pipeline 2 and the "explode" approach before (left) and after (right) the k-fold split, respectively.

Table 4
The number of examples in each resource dataset.

Dataset	Original size	After preprocessing	After "explode"
WordNet 3.0	65,018	59,666	387,814
Wikipedia	33,547	30,603	371,766

5.3. Study case 3: Cross-resource experiments

In the previous two study cases, we proved that informal definitions are the best way to represent domain entities in classifying domain entities into top-level ontology concepts and show that our hypothesis is also valid for languages other than English, with some considerations in using the "explode" approach after the k-fold split. However, we used only the BERT-Base language model in these experiments since it is light and fast enough to generate several experiments to validate our hypothesis. Also, the results are drastically affected by the "explode" approach. In this context, if we use two datasets with informal definitions from different resources, one as the training sample and the other as the test sample, we can avoid using a k-fold split and the choice of which "explode" approach should be used. From that, we can argue that different resources have different informal definitions for the domain entities. Thus, by performing an experiment in a cross-resource classification scenario, we can evaluate the performance of both "explode" approaches at the same time, where we aim to classify a domain entity from an informal definition with completely different definiens from those used to train the classifier and also having other informal definitions of the same domain entity in the training sample.

In this study case, we evaluated several language models for both pipelines described in Section 4.4. For Pipeline 1, we used the BERT-Base, GPT2, and GPT2 with GPTQ quantization, fine-tuned using 4 and 10 epochs. In Pipeline 2, we used the BERT-Base, BERT-Large, ROBERTA-Base, ALBERT-Base, GPT2, GPT2 with GPTQ quantization, Gemma2B, Gemma7B with 4bit quantization, Mistral7B with 4bit quantization, T5, and BART. Also, for Pipeline 2, we only used the KNN classifier because it is the one that best fits our hypothesis, as presented in the previous study case. In addition, Table 4 presents the number of examples of each evaluated dataset after performing preprocessing and "explode" steps. From that, we used the informal definitions from WordNet 3.0 to train the pipelines and the informal definitions from Wikipedia to evaluate the pipelines.

Figure 11 presents the overall performance of all evaluated language models in their designated pipelines. The best approach was combining the quantized version of Mistral7B in 4bit with the KNN classifier, achieving 84% of macro F1-score. Although using quantized versions of the language models allows us to evaluate bigger language models faster, this approach can drastically reduce the performance of language models. This can be seen by comparing the results of models that used Gemma2B and the quantized version of Gemma7B, where the first achieves 78% of macro F-score and the second achieves a drastically poor performance of 46%. The same issue can also be viewed with the GPT2 and GPT-GPTQ experiments for both pipelines. In addition, the encoder-decoder architectures of BART-Base and T5-Base achieve a medium range of scores of 66% and 68%, respectively. Also, the BERT variations ROBERTA and ALBERT achieve worse scores than the BERT model, with 62% and 63%, respectively. The bigger version of BERT, BERT-Large, achieves a slightly better score than BERT-Base, with 0.2% improvement, suggesting that using larger models does not result in a notable improvement in results and may not compensate for the extra computational cost.

Based on Figure 11, we can also compare both pipelines using the same language models. In this context, using BERT-Base with a KNN classifier, we obtained the best macro F-score result, with 17.5% and

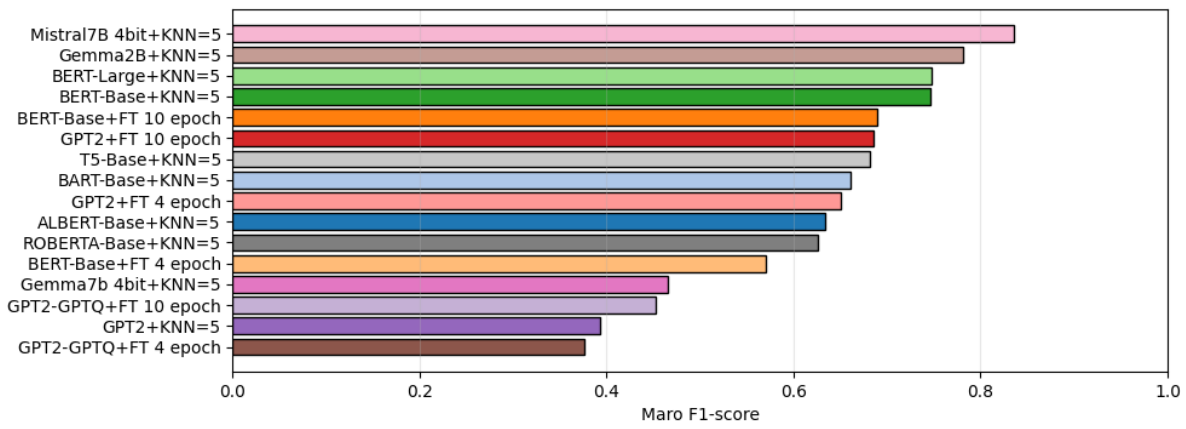

Fig. 11. Macro F-score results for each evaluated language model across the proposed pipelines.

5.6% improvement over the fine-tuned versions using 4 and 10 epochs, respectively. On the other hand, fine-tuning the GPT2 model for 10 epochs to classify domain entities into top-level concepts using informal definitions as input achieves the best macro F-score results, with 26.6% and 3.5% improvement over the GPT2+KNN and GPT2 model fine-tuned for 4 epochs. A question that remains from this analysis is: what happens if we fine-tune these models for more epochs? Figure 12 presents a detailed overview of the macro F-score and the training and evaluation curves over the 10 epochs. In this image, we can see that the F-score for both BERT and GPT2 models begins to stagnate as the epochs increase, and we can even use fewer epochs and obtain the same F-score value for both cases.

This case study demonstrates an effective approach to confirm our hypothesis that the informal definitions represent semantic information that allows domain entities to be related to top-level ontology

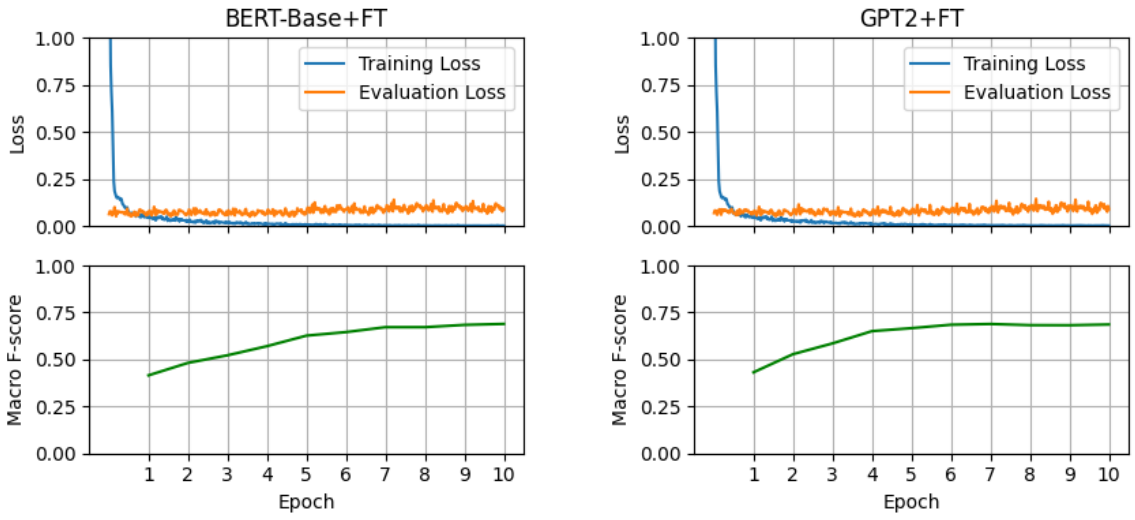

Fig. 12. The relation between the number of epochs used to train the models and respective macro F-score, training loss, and evaluation loss curves.

concepts. Also, it illustrates that employing a KNN classifier with contextual embeddings derived from informal definitions yields better results than fine-tuning methods. This outcome supports the notion that the distributional properties of informal definitions encapsulate the top-level ontology concepts associated with domain entities. Furthermore, the enhancement of this process by state-of-the-art language models such as Mistral and Gemma indicates that the classification of domain entities into top-level ontology concepts via informal definitions and contextual embeddings benefits from advancements in language model improvements.

6. Conclusion

Drawing from the detailed experimentation of classifying domain entities into top-level ontology concepts through informal definitions, this study explains a novel approach leveraging multi-label, multi-resource, and multi-language dimensions of the problem. Our study started by proposing an approach for aligning the OntoWordNet and Babel and a systematic methodology to extract the multi-label, multi-language, and multi-resource datasets from this alignment, addressing and solving all the limitations found during this process. Also, by harnessing state-of-the-art language models, we developed two distinct pipelines: one focusing on fine-tuning pre-trained language models specifically for our classification task and another utilizing these models to generate embeddings for classical machine learning classifiers. This dual approach allowed us to navigate the complexities of the distributional hypothesis effectively, validating our hypothesis that the informal definitions represent semantic information that allows domain entities to be related to top-level ontology concepts in several ways. In addition, our findings underscore the significant impact of using informal definitions to represent domain entities in relation to other representation approaches, demonstrating their intrinsic value in capturing the nuances necessary for accurately representing domain entities through contextual embedding from language models. The best results were achieved using the K-Nearest Neighbor approach with embeddings from the Mistral7B language model, highlighting the effectiveness of this classical machine-learning approach over the expensive computational cost of fine-tuning. This work not only supports our hypothesis but also emphasizes the potential for automated approaches to assist ontology engineers during the development process. Furthermore, our experiments across different languages and resources demonstrate the versatility and robustness of our approach. In future works, we will aim to expand the datasets for other top-level concepts not covered by the OntoWordNet. Also, we will perform new experiments by translating the informal definitions in English to different languages to use the same data across all multi-language experiments.

Acknowledgments

Research supported by Higher Education Personnel Improvement Coordination (CAPES), code 0001, Brazilian National Council for Scientific and Technological Development (CNPq), and Petrobras.

References

Arp, R., Smith, B. & Spear, A.D. (2015). *Building ontologies with basic formal ontology*. Mit Press.

Babaei Giglou, H., D'Souza, J. & Auer, S. (2023). LLMs4OL: Large language models for ontology learning. In *International Semantic Web Conference* (pp. 408–427). Springer.

Borgo, S., Ferrario, R., Gangemi, A., Guarino, N., Masolo, C., Porello, D., Sanfilippo, E.M. & Vieu, L. (2022). DOLCE: A descriptive ontology for linguistic and cognitive engineering. *Applied ontology*, *17*(1), 45–69.

Chen, J., He, Y., Geng, Y., Jiménez-Ruiz, E., Dong, H. & Horrocks, I. (2023). Contextual semantic embeddings for ontology subsumption prediction. *World Wide Web*, 1–23.

Cicconeto, F., Vieira, L.V., Abel, M., dos Santos Alvarenga, R., Carbonera, J.L. & Garcia, L.F. (2022). GeoReservoir: An ontology for deep-marine depositional system geometry description. *Computers & Geosciences*, *159*, 105005.

Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Gangemi, A., Navigli, R. & Velardi, P. (2003). The OntoWordNet Project: Extension and Axiomatization of Conceptual Relations in WordNet. In R. Meersman, Z. Tari and D.C. Schmidt (Eds.), *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE* (pp. 820–838). Berlin, Heidelberg: Springer Berlin Heidelberg.

Gangemi, A., Guarino, N., Masolo, C., Oltramari, A. & Schneider, L. (2002). Sweetening ontologies with DOLCE. In *International Conference on Knowledge Engineering and Knowledge Management* (pp. 166–181). Springer.

Guarino, N. (1998). *Formal ontology in information systems: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy* (Vol. 46). IOS press.

Guarino, N., Oberle, D. & Staab, S. (2009). What Is an Ontology? In *Handbook on Ontologies*.

Guizzardi, G., Botti Benevides, A., Fonseca, C.M., Porello, D., Almeida, J.P.A. & Prince Sales, T. (2022). UFO: Unified foundational ontology. *Applied ontology*, *17*(1), 167–210.

Harris, Z.S. (1954). Distributional structure. *Word*, *10*(2-3), 146–162.

He, Y., Chen, J., Antonyrajah, D. & Horrocks, I. (2022). BERTMap: a BERT-based ontology alignment system. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 36, pp. 5684–5691).

He, Y., Chen, J., Dong, H., Horrocks, I., Allocca, C., Kim, T. & Sapkota, B. (2023). DeepOnto: A Python Package for Ontology Engineering with Deep Learning.

Jackson, R., Matentzoglu, N., Overton, J.A., Vita, R., Balhoff, J.P., Buttigieg, P.L., Carbon, S., Courtot, M., Diehl, A.D., Dooley, D.M., et al. (2021). OBO Foundry in 2021: operationalizing open data principles to evaluate ontologies. *Database*, *2021*.

Jiang, A.Q., Sablayrolles, A., Roux, A., Mensch, A., Savary, B., Bamford, C., Chaplot, D.S., de las Casas, D., Hanna, E.B., Bressand, F., Lengyel, G., Bour, G., Lample, G., Lavaud, L.R., Saulnier, L., Lachaux, M.-A., Stock, P., Subramanian, S., Yang, S., Antoniak, S., Scao, T.L., Gervet, T., Lavril, T., Wang, T., Lacroix, T. & Sayed, W.E. (2024). Mixtral of Experts.

Khadir, A.C., Aliane, H. & Guessoum, A. (2021). Ontology learning: Grand tour and challenges. *Computer Science Review*, *39*, 100339.

Kulvatunyou, B., Drobnjakovic, M., Ameri, F., Will, C. & Smith, B. (2022). The Industrial Ontologies Foundry (IOF) Core Ontology. Formal Ontologies Meet Industry (FOMI) 2022, Tarbes, FR. https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=935068.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P. & Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Lopes, A., Carbonera, J.L., Schimidt, D. & Abel, M. (2022). Predicting the top-level ontological concepts of domain entities using word embeddings, informal definitions, and deep learning. *Expert Systems with Applications*, *203*, 117291.

Lopes, A., Carbonera, J., Schmidt, D., Garcia, L., Rodrigues, F. & Abel, M. (2023). Using terms and informal definitions to classify domain entities into top-level ontology concepts: An approach based on language models. *Knowledge-Based Systems*, *265*, 110385.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S. & Dean, J. (2013a). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111–3119).

Mikolov, T., Chen, K., Corrado, G. & Dean, J. (2013b). Efficient estimation of word representations in vector space. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Miller, G.A. (1995). WordNet: a lexical database for English. *Communications of the ACM*, *38*(11), 39–41.

Navigli, R. & Velardi, P. (2004). Learning Domain Ontologies from Document Warehouses and Dedicated Web Sites. *Computational Linguistics*, *30*, 151–179. https://api.semanticscholar.org/CorpusID:2453822.

Navigli, R., Bevilacqua, M., Conia, S., Montagnini, D. & Cecconi, F. (2021). Ten Years of BabelNet: A Survey. In *IJCAI* (pp. 4559–4567).

Niles, I. & Pease, A. (2001). Towards a standard upper ontology. In *Proceedings of the international conference on Formal Ontology in Information Systems-Volume 2001* (pp. 2–9).

Otte, J.N., Beverley, J. & Ruttenberg, A. (2022). BFO: Basic formal ontology. *Applied ontology*, *17*(1), 17–43.

Pennington, J., Socher, R. & Manning, C.D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532–1543).

Prestes, E., Carbonera, J.L., Fiorini, S.R., Jorge, V.A., Abel, M., Madhavan, R., Locoro, A., Goncalves, P., Barreto, M.E., Habib, M., et al. (2013). Towards a core ontology for robotics and automation. *Robotics and Autonomous Systems*, *61*(11), 1193–1204.

Qu, Y., Perrin, M., Torabi, A., Abel, M. & Giese, M. (2024). GeoFault: A well-founded fault ontology for interoperability in geological modeling. *Computers & Geosciences*, *182*, 105478.

Radford, A., Narasimhan, K., Salimans, T. & Sutskever, I. (2018). Improving language understanding by generative pre-training. *URL https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language understanding paper.pdf*.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W. & Liu, P.J. (2023). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer.

Reimers, N. & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks.

Robinson, R. (1950). *Definition*. Oxford: Clarendon Press.

Rodrigues, F.H., Lopes, A.G., dos Santos, N.O., Garcia, L.F., Carbonera, J.L. & Abel, M. (2023). On the Use of ChatGPT for Classifying Domain Terms According to Upper Ontologies. In *International Conference on Conceptual Modeling* (pp. 249–258). Springer.

Sanh, V., Debut, L., Chaumond, J. & Wolf, T. (2020). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter.

Seppälä, S., Ruttenberg, A., Schreiber, Y. & Smith, B. (2016). Definitions in Ontologies. *Cahiers de Lexicologie*, *2016*, 173–205.

Studer, R., Benjamins, V.R. & Fensel, D. (1998). Knowledge engineering: Principles and methods. *Data & knowledge engineering*, *25*(1-2), 161–197.

Suárez-Figueroa, M.C., Gómez-Pérez, A. & Fernández-López, M. (2011). The NeOn methodology for ontology engineering. In *Ontology engineering in a networked world* (pp. 9–34). Springer.

Suárez-Figueroa, M.C., Gómez-Pérez, A. & Fernandez-Lopez, M. (2015). The NeOn Methodology framework: A scenario-based methodology for ontology development. *Applied ontology*, *10*(2), 107–145.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E. & Lample, G. (2023). LLaMA: Open and Efficient Foundation Language Models.

Wong, W., Liu, W. & Bennamoun, M. (2012). Ontology learning from text: A look back and into the future. *ACM computing surveys (CSUR)*, *44*(4), 1–36.