

Large Language Models for Creating BFO-Compliant Taxonomies from Structured Texts: A Demonstration on Tensile Testing Taxonomy

Hossein Beygi Nasrabadi^{1,*}, Farzaneh Mahdavi², Jörg Waitelonis¹ and Harald Sack^{1,3}

¹Information Service Engineering, FIZ Karlsruhe - Leibniz Institute for Information Infrastructure, Hermann-von-Helmholtz-Platz 1, Eggenstein-Leopoldshafen, 76344, Germany

²Department of Computer Science, Technical University of Darmstadt, Karolinenplatz 5, 64289 Darmstadt, Germany

³Institute of Applied Informatics and Formal Description Methods, Karlsruhe Institute of Technology, Hermann-von-Helmholtz-Platz 1, Eggenstein-Leopoldshafen, 76344, Germany

Abstract

Developing formal ontologies remains a labor-intensive task that demands scarce domain expertise. Although large language models (LLMs) offer new opportunities to automate this process, ensuring their outputs comply with top-level ontologies still remains an open challenge. This paper introduces LLM4BFO, a pipeline for generating Basic Formal Ontology (BFO)-aligned OWL taxonomies from structured text resources, including technical documents, recipes, and standards. The pipeline employs few-shot, rule-augmented, and BFO-grounded prompting to guide term and definition extraction from text, BFO-compliant class hierarchy construction, Aristotelian definition generation, and OWL serialization of resulting taxonomies. LLM4BFO is evaluated on two structured text sources, an egg-boiling recipe and the ISO 6892-1 tensile testing standard, with results compared against human-created ontologies, showing high lexical similarity of labels (F_1 -score of 0.96), meaningful semantic definition preservation (77%), and reliable BFO hierarchy alignment (F_1 -score of 0.96).

Keywords

Basic Formal Ontology, Large Language Models (LLMs), Ontology Engineering, Tensile Testing

1. Introduction

During recent years, multiple works have examined the application of large language models (LLMs) to ontology engineering, which could be generally divided into ontology development, ontology alignment, and ontology evaluation. Regarding ontology development, LLMs create initial OWL ontologies based on natural language text, including competency questions (CQs), user stories, or domain texts via memoryless CQ-by-CQ approach or Ontogenia [1, 2]. In addition, LLMs may perform axiom and terminology extraction from text and offer candidates of classes, properties, relationships, and definitions, as demonstrated in systems such as Protégé integrated pipelines [3], NeOn-GPT [4], LLMs4Life [5], or DRAGON-AI pipelines with retrieval augmented generation (RAG) [6]. As far as ontology alignment is concerned, LLMs help to map and integrate different ontologies by enhancing semantic recall, along with other symbolic tools, e.g., MapperGPT [7], OntoTune [8] or LLMs4OM [9]. Regarding ontology evaluation, LLMs support CQs answering [2], human-in-the-loop refinement and reuse of patterns, as well as error identification [1], and comparison of inference of axioms with human judgement [10], while their applicability for functional adequacy assessment is still relatively undeveloped [9].

Despite being quite successful in assisting ontology engineering, LLMs are still incapable of substituting the expert-driven approaches entirely due to multiple challenges. The generated ontologies tend to contain inconsistencies, hallucinations, shallow top-level ontology alignment, and low reliability

SeMatS 2026: The 3rd International Workshop on Semantic Materials Science co-located with the 25th International Semantic Web Conference (ISWC 2026), October 25th, 2026, Bari, Italy

*Corresponding author.

✉ Hossein.Beygi_Nasrabadi@fiz-Karlsruhe.de (H. Beygi Nasrabadi)

ORCID 0000-0002-3092-0532 (H. Beygi Nasrabadi); 0000-0001-7192-7143 (J. Waitelonis); 0000-0001-7069-9804 (H. Sack)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

[1, 2, 11]. Issues such as incomplete or biased knowledge, poor structure, prompt sensitivity, and non-determinism hinder quality, scalability, and reproducibility [11, 12]. Therefore, expert validation remains essential, and LLMs should be used as supportive tools within supervised, quality-controlled workflows rather than fully automated solutions [2].

This study hypothesizes that grounding LLM-developed ontologies in top-level ontologies substantially improves their quality and interoperability. Top-level ontologies provide a shared semantic foundation, enforce logical consistency, reduce hallucinations and structural errors, and enable cross-domain integration and reuse [13]. Especially, Basic Formal Ontology (BFO) is internationally recognized as the standardized top-level ontology, offering a highly general and extensible structure adopted across hundreds of projects in medicine, industrial manufacturing, and defense, making it the most suitable foundation for this work [14, 13].

2. Related works

To date, there is no comprehensive research demonstrating the use of LLMs for developing ontologies aligned with top-level ontologies. Existing works are limited to the two ontology development sub-tasks:

I- LLM for Aristotelian definition of classified entities: The Aristotelian definition is the standard approach for defining ontological entities. It defines a concept through its genus (parent class) and differentia (distinguishing characteristics), typically following the form “x is a y (genus) that z (differentia)” [15]. Several studies have explored LLM-based generation of Aristotelian definitions. Procko et al. [16] evaluated LLM-generated OWL definitions for BFO compliance, while Joachimiak et al. [17] used a human-in-the-loop LLM process to refine and generate definitions for the Artificial Intelligence Ontology. Benson et al. [12] developed "My Ontologist", a GPT-based chat interface for generating BFO-conformant definitions. Although it produced valid RDF/OWL triples, the system often generated vague definitions, incorrect hierarchies, hallucinated properties, BFO violations, and invalid IRIs. These findings indicate that LLM-generated ontology definitions still require rule-based constraints, ontology validation, and expert oversight [12].

II- LLM for classification of defined entities: Abel group performed some research on LLM classifying domain entities into top-level ontology concepts based on the informal definitions of the terms [18, 19, 20, 21, 22]. Their approach trains LLM-based classifiers on top-level ontology categories and definitions, then predicts classifications for entities extracted from existing ontologies [18]. Experiments on classifying OntoWordNet and BabelNet datasets based on DOLCE-Lite-Plus (DLP) top-level ontology achieved the best results with Mistral embeddings and K-Nearest Neighbor classification method [19, 20]. For the use case of entities classification based on BFO top-level ontology, the BERT language model was used for classifying classes of 81 ontologies, achieving up to 90% F-scores for "certain" BFO classes [21]. Nevertheless, these approaches face several limitations: classification is influenced by domain specificity, where definitions from specialized ontologies can lead to misclassification due to overlapping embeddings [22]; reuse of entities and definitions across BFO-aligned ontologies complicates fair evaluation; Aristotelian-style definitions often explicitly state the top-level class; and the approach evaluates classification of pre-defined entities rather than supporting newly defined ones.

Despite these contributions, existing approaches remain fragmented and focus on isolated subtasks of ontology engineering. Most work on classifying domain entities into top-level ontology concepts relies on pre-existing classes with available definitions, rather than supporting the classification of newly introduced terms. Another key gap is the need for faithful evaluation and systematic technical comparison of LLM-generated outputs against gold-standard ontologies. Overall, these methods are still far from enabling reliable, end-to-end generation of OWL-serialized, top-level-ontology-aligned ontologies directly from textual resources.

3. Methodology

LLM4BFO is an open-source system for LLM-driven BFO-aligned ontology development. The complete LLM4BFO architecture is described in [23]; also, all source code, prompts, and evaluation scripts can be found in the project’s repository [24]. In brief, the pipeline accepts ontology terms and definitions from heterogeneous sources (PDF, XLSX, TTL, OWL), together with ontology metadata and an LLM provider API key, and produces a validated BFO-compliant ontology in Turtle serialization. The generation pipeline follows a few-shot, rule-augmented, instruction-based prompting method structured into four steps.

i) Structured input prompting: Domain terms and definitions are passed to the model as structured JSON (label, definition), which improves vocabulary interpretation and consistent class mapping.

ii) Embedded reference grounding: The prompt embeds a compact BFO [25] reference specification (class descriptions, taxonomic relations, and classification decision rules) that functions as a rule-encoded decision tree guiding BFO category assignment independently of the model’s internal knowledge.

iii) Constrained instruction prompting: Explicit ontology engineering rules are embedded in the prompt, covering class creation (one owl:Class per concept, unique IRIs, rdfs:label), hierarchy design (single BFO superclass, rdfs:subClassOf relations, local hierarchy consistency), and class definitions (both source an LLM-created Aristotelian definition via skos:definition).

iv) Deterministic output constraint prompting: The prompt enforces valid Turtle serialization through required prefix declarations, a complete ontology header, an explicit owl:imports of BFO, and consistent RDF triple structure. The generated TTL file is then validated using RDFLib to confirm syntactic correctness and BFO grounding.

4. BFO-compliant taxonomy creation from structured texts

This section investigates the capability of LLM4BFO to generate BFO-compliant ontologies from structured textual sources. By "structured," we mean documents (such as PDFs) that contain clearly identifiable sections in which terms and their corresponding definitions can be explicitly located. We believe that class labels alone are insufficient for ontology construction, and access to definitions enables the system to produce semantically meaningful ontology structures. A structured text containing domain-specific terminology and definitions supports ontology generation in two ways: it provides a well-defined domain scope that helps the LLM better interpret the semantics of the domain, and it improves the generation of semantically accurate definitions and lexically appropriate class labels.

Two use cases of structured text inputs were employed in this section. In the first use case, a short document on "egg boiling recipe" was used (Fig. 1). We intentionally constructed this document to ensure accessibility for readers from different domains, providing a simple and universally understandable demonstration of ontology generation from text. The document is a one-page structured text containing several sections, one of which explicitly lists terms and definitions related to egg boiling (for example, "egg," "boiling," and "peelability"). The document contains approximately ten main terms, one of which include three sub-terms, yielding an expected total of 13 terms and definitions for a correctly performing extraction system.

Using LLM4BFO with the GPT-5 model, the egg boiling taxonomy in Turtle serialization format was generated from this document. Through the interface (Fig. 1), the PDF was selected as the input source, and a small set of basic ontology metadata was provided (including title, preferred label, and version information). The generated taxonomy is syntactically valid and can be successfully opened and processed in Protégé. As shown in Fig. 1, it correctly imports BFO and includes ontology-level annotations derived from the user-provided metadata. The Protégé class hierarchy of the LLM-generated taxonomy is also presented on the right side of Fig. 2.

An important challenge in this study is the absence of suitable reference datasets, specifically, BFO-compliant ontologies constructed by ontology engineers directly and exclusively from textual documents. It should be noted that, in our other research [23], we evaluated the BFO-compliant classification of

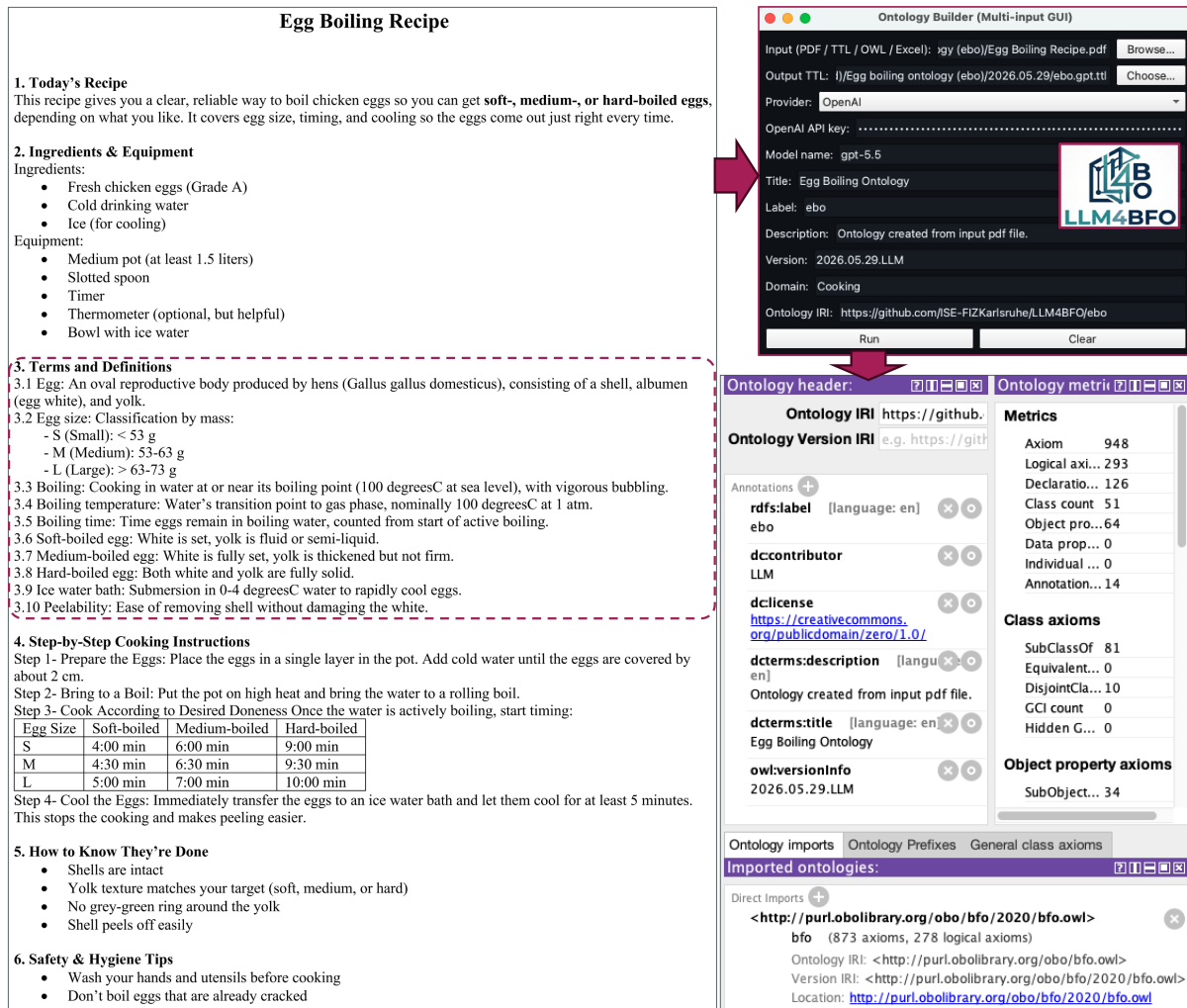
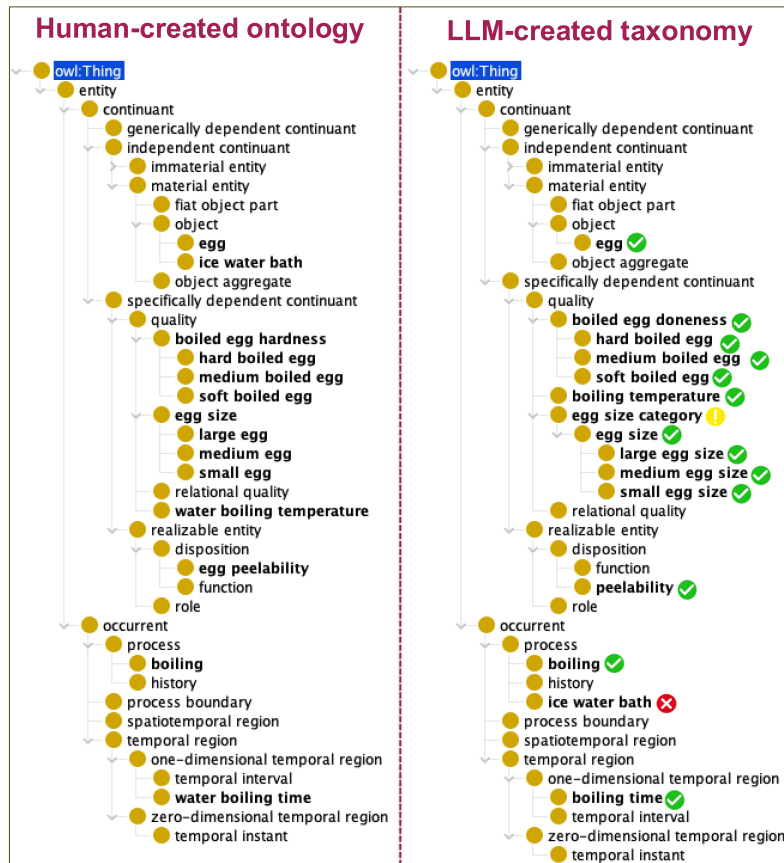


Figure 1: Egg boiling recipe as the structured text input to LLM4BFO, and the resulting BFO-compliant taxonomy visualized in Protégé.

entities by regenerating 16 BFO-aligned ontologies, where the resulting F_1 -score of above 0.9 suggested strong performance in BFO-compliant hierarchy construction. However, for the purpose of evaluating LLM-created taxonomies from textual resources, some comparable human-generated ontologies were created in this research prior to LLM-based taxonomy generation. The ontology engineers were intentionally constrained to use only the provided text and to model the ontology according to BFO principles, to enable a more faithful comparative analysis. For example, the class hierarchy of the human-created egg boiling ontology is shown on the left side of Fig. 2.

The evaluation scope of this study is to analyze how LLM4BFO supports taxonomy creation from structured texts, specifically, whether the generated ontology classes correspond to the expected concepts in the source text, whether the generated labels and definitions are appropriate, and whether the resulting class hierarchy follows BFO alignment. It should be noted that the scope of this research is limited to taxonomies: the generated Turtle files contain only labeled and defined classes with BFO-compliant hierarchies, without properties or axioms. Therefore, the evaluation of more advanced ontology elements, such as properties and logical axioms, is outside the scope of the present study.

A first inspection of Fig. 2 suggests that the LLM-generated taxonomy successfully extracts all relevant terms from the source text, consistent with the human-created ontology. Both taxonomies contain one class per relevant term identified in the text. In addition, local superclass structures were added in both cases where additional abstraction levels were required for organizing semantically related concepts. The generated ontology also correctly imports BFO and organizes the classes according to an



appropriate BFO hierarchy. To facilitate visual comparison, color-coded indicators are used in Fig. 2; classes that match perfectly between the two taxonomies are highlighted in green, additional local classes introduced in the LLM-generated taxonomy are highlighted in yellow, and classes present in both taxonomies but assigned to different hierarchical positions are highlighted in red.

A corresponding quantitative analysis is presented in Table 1. It compares the terms extracted from the reference text with classes in the human- and LLM-created ontologies, and provides a comparison between the class labels, definitions, and BFO hierarchies. The source text contains 13 primary terms. As shown in Table 1, the human-created ontology contains 14 classes, while the LLM-created one contains 15. The additional classes in both cases reflect superclasses introduced to better organize semantically related concepts. For example, given the terms "hard-boiled egg," "medium-boiled egg," and "soft-boiled egg" in the source text, the human-created ontology introduced "Boiled Egg Hardness" as a superclass, while the LLM one introduced "Boiled Egg Doneness" for the same purpose. Importantly, all terms from the source text were successfully extracted by LLM4BFO and correspond to matching classes in the human-created ontology.

A closer inspection of class labels reveals that both ontologies frequently use nearly identical labels (for example, "Boiling," "Egg," and "Egg Size"), reflecting a shared tendency to remain faithful to the terminology of the source text. In some cases, however, both the human and the LLM intentionally modified certain labels, primarily because the original terms were insufficiently descriptive as ontology class names. For instance, the source text uses shorthand notation such as "S (small)," "M (medium)," and "L (large)" for subclasses of "Egg Size," which are unsuitable as standalone class names; both taxonomies replaced these with more explicit labels such as "Small Egg," "Medium Egg," and "Large Egg". Minor semantic refinements were also applied in both cases to improve label clarity and uniqueness; for example, "Peelability" was refined to "Egg Peelability," and the human-generated ontology further refined "Boiling Temperature" and "Boiling Time" to "Water Boiling Temperature" and "Water Boiling Time".

Table 1

Comparison of class labels, definitions, and BFO hierarchy assignments between the human-generated and LLM-generated taxonomies derived from the egg boiling recipe document.

#	Text	Human-created ontology			LLM-created taxonomy		
		Label	Definition	BFO parent	Label	Definition	BFO parent
1	Boiling	boiling	Cooking in water at or near its boiling point (100°C at sea level), with vigorous bubbling.	bfo:0000015	boiling	A boiling is a process that cooks something in water at or near its boiling point with vigorous bubbling.	bfo:0000015
2	Egg	egg	An oval reproductive body produced by hens (<i>Gallus gallus domesticus</i>), consisting of a shell, albumen (egg white), and yolk.	bfo:0000030	egg	An egg is an object that is an oval reproductive body produced by hens and consists of a shell, albumen, and yolk.	bfo:0000030
3	Peelability	egg peelability	Ease of removing shell without damaging the white.	bfo:0000016	peelability	A peelability is a disposition that is realized in removing the shell from an egg without damaging the white.	bfo:0000016
4	Egg size	egg size	Classification by mass	bfo:0000019	egg size	An egg size is an egg size category that classifies eggs by mass into S, M, and L grades.	bfo:0000019
5	Hard-boiled egg	hard boiled egg	Both white and yolk are fully solid.	bfo:0000019	hard boiled egg	A hard boiled egg is a boiled egg doneness that has both white and yolk fully solid.	bfo:0000019
6	Ice water bath	ice water bath	Submersion in 0–4°C water to rapidly cool eggs.	bfo:0000030	ice water bath	An ice water bath is a process that submerges eggs in 0–4°C water to rapidly cool them.	bfo:0000015
7	L (Large)	large egg	>63–73 g	bfo:0000019	large egg size	A large egg size is an egg size that applies to eggs with a mass greater than 63 g and up to 73 g.	bfo:0000019
8	Medium-boiled egg	medium boiled egg	White is fully set, yolk is thickened but not firm.	bfo:0000019	medium boiled egg	A medium boiled egg is a boiled egg doneness that has a fully set white and a thickened but not firm yolk.	bfo:0000019
9	M (Medium)	medium egg	53–63 g	bfo:0000019	medium egg size	A medium egg size is an egg size that applies to eggs with a mass from 53 g to 63 g.	bfo:0000019
10	S (Small)	small egg	<53 g	bfo:0000019	small egg	A small egg size is an egg size that applies to eggs with a mass below 53 g.	bfo:0000019
11	Soft-boiled egg	soft boiled egg	White is set, yolk is fluid or semi-liquid.	bfo:0000019	soft boiled egg	A soft boiled egg is a boiled egg doneness that has a set white and a fluid or semi-liquid yolk.	bfo:0000019
12	Boiling temperature	water boiling temperature	Water’s transition point to gas phase, nominally 100°C at 1 atm.	bfo:0000019	boiling temperature	A boiling temperature is a quality that characterizes water at its nominal transition to gas phase at 100°C and 1 atm.	bfo:0000019
13	Boiling time	water boiling time	Time eggs remain in boiling water, counted from start of active boiling.	bfo:0000038	boiling time	A boiling time is a one dimensional temporal region that spans the interval during which eggs remain in boiling water starting from active boiling.	bfo:0000038
14		boiled egg hardness	A quality that represents the hardness degree of the boiled eggs.	bfo:0000019	boiled egg doneness	A boiled egg doneness is a quality that characterizes a boiled egg by the degree to which its white and yolk are set.	bfo:0000019
15					egg size category	An egg size category is a quality that groups hen eggs according to bounded mass grades.	bfo:0000019

to better align the labels with their intended definitions. Regarding the additional local superclasses introduced in both ontologies, some appear redundant (such as "Egg Size Category"), while others represent meaningful hierarchical improvements, with lexically close labels across both taxonomies (such as "Boiled Egg Hardness" or "Boiled Egg Doneness"). Overall, these results indicate that LLM4BFO remains faithful to the source text while refining labels where the original terminology is insufficiently descriptive or semantically ambiguous, with labeling performance comparable to that of the human

Table 2

Quantitative evaluation of lexical label similarity, semantic definition similarity, and BFO hierarchy alignment between the human-generated and LLM-generated taxonomies.

Matched classes (#)	Label lexical similarity (F_1)	Definitions semantic similarity (%)	Classes BFO hierarchy match (F_1)	Local parent hierarchy match (F_1)
14	0.965	77.08	0.963	0.833

engineers.

To quantify label similarity between the two ontologies, a lexical similarity analysis was performed using the Jaro–Winkler metric (jellyfish library). Candidate pairs above a similarity threshold of 0.8 were retained, and a greedy one-to-one matching strategy was applied to determine final correspondences, with precision, recall, and F_1 -score used to quantify lexical agreement. The resulting F_1 -score of 0.965 (Table 2) confirms that the class labels generated by LLM4BFO are lexically very close to those produced by the human ontology engineers.

A similar comparison can be made for the class definitions. The human-created ontology preserved the original definitions exactly as they appeared in the source text, copying them directly without modification. However, many of these source definitions are not fully appropriate as ontology definitions. For example, the definition of "Egg Size" in the source text is phrased as "classification by mass" rather than as a proper ontology definition, and the definitions of "Small Egg," "Medium Egg," and "Large Egg" are semantically incomplete. The input document therefore provides terms with associated descriptions that are not always usable ontological definitions. In contrast, LLM4BFO reformulates and improves definitions based on both the class label and the source text description, producing definitions in Aristotelian form. For example, "Small Egg" is defined as: "A small egg size is an egg size that applies to eggs with a mass below 53 g", a definition that specifies both the parent category and the distinguishing property, making it more suitable for ontology engineering. It is also worth noting that the LLM-generated taxonomy preserves both the original and generated Aristotelian definition versions as skos:definition annotations.

To quantitatively compare definitions across the two ontologies, a semantic similarity analysis was performed. Textual definitions were embedded using a SentenceTransformer model (all-MiniLM-L6-v2), and cosine similarity between embeddings was used to measure semantic equivalence, with a threshold of 0.75 applied to identify matched pairs. The analysis compares the Aristotelian definitions generated by LLM4BFO against the original source definitions for all matched ontology classes. As shown in Table 2, the average semantic similarity is approximately 77%, indicating that the LLM generally preserves the original semantics of the source definitions while improving their ontological quality. The observed differences are primarily attributable to cases in which the source definitions were incomplete or unsuitable (e.g., "Egg Size," "Small Egg"), requiring reformulation and enrichment by LLM.

Regarding class hierarchies, all 14 matched classes between the two taxonomies were compared with respect to their hierarchical structure. A "hierarchy match" here refers to whether classes are assigned to the same top-level BFO parent category and whether their local hierarchical organization is consistent. Based on the color-coded indicators in Fig. 2, most classes are hierarchically aligned across the two taxonomies. As with the human-generated ontology, LLM4BFO attempts to preserve hierarchical structures already implicitly present in the source text, for example, the concept "Egg Size" together with its sub-categories (small, medium, and large eggs) was reconstructed as a hierarchy in both taxonomies. Additional superclasses were also introduced in both cases where concept categorization was required: the classes "Soft-Boiled Egg," "Medium-Boiled Egg," and "Hard-Boiled Egg" were grouped under "Boiled Egg Hardness" in the human-created ontology and under "Boiled Egg Doneness" in the LLM-created one. Overall, nearly all classes are successfully matched between the two ontologies. The only discrepancy is the additional class "Egg Size Category" appearing in the LLM-created ontology, which, given the existing class "Egg Size," represents a redundant abstraction that could be simplified or removed.

The single hierarchical mismatch between the two ontologies concerns the class "Ice Water Bath," indicated by the red marker in Fig. 2. In the human-generated ontology, this concept is modeled as a `bfo:Object`, whereas in the LLM-generated ontology it is modeled as a `bfo:Process`. A closer analysis traces this disagreement to the generated definition, which describes "submersion in 0–4°C water to rapidly cool eggs", a description that semantically resembles a process rather than a physical object, leading the LLM to classify it accordingly. This case illustrates an inconsistency between the class label and its associated definition in the source text: the label refers to an object, while the definition describes a process. The human engineers prioritized the label interpretation, whereas the LLM prioritized the semantic content of the definition.

In Table 2, the "classes BFO hierarchy match" metric was used to evaluate, for each matched class pair, whether the BFO parent of the LLM-created class corresponds to that of the human-created one. The resulting F_1 -score of 0.963 indicates a very high degree of structural agreement between the two ontologies. The "local parent hierarchy match" metric was additionally used to assess whether local hierarchical structures are preserved consistently, for example, whether "Small Egg," "Medium Egg," and "Large Egg" are organized under equivalent local parent classes in both ontologies. The resulting F_1 -score for this metric is 0.833, with the primary source of disagreement being the additional superclass "Egg Size Category" introduced in the LLM-generated ontology, which creates a slight structural deviation. Overall, these results demonstrate that the class hierarchy generated by LLM4BFO from structured textual input is highly suitable and strongly aligned with BFO. It should be noted, however, that the current implementation supports only relatively basic hierarchical alignment; more advanced modeling scenarios, such as multiple inheritance and polyhierarchical structures, are not yet addressed.

As the second use case, we evaluated the suitability of ISO standard texts as structured input for BFO-compliant ontology generation. ISO standards are particularly well-suited for this purpose, as they systematically list terms and definitions organized by domain topic, precisely the structure required by LLM4BFO. Beyond their structural properties, standards represent authoritative sources of domain terminology established through expert consensus, offering domain-specific precision and ensuring that concepts are described consistently and unambiguously across contexts. Their wide adoption across industries supports cross-industry applicability and improves interoperability between systems and organizations. Furthermore, standard texts are stable, regularly maintained, and carefully documented, making them a reliable foundation for long-term ontology engineering. Overall, using standards as input for ontology development ensures that the resulting models are scientifically grounded and aligned with industrial practice, increasing their acceptance, usability, and long-term relevance [26].

The ISO standard documents naturally satisfy the primary requirement of the LLM4BFO framework: the availability of clearly identifiable terms and corresponding definitions within a structured textual format. One practical limitation, however, is that ISO standards are not freely accessible, full documents typically require purchase, and while introductory sections and selected terminology may be partially available on the ISO website, complete terminology and definition sections are generally restricted due to licensing and copyright. From a copyright perspective, verbatim and extensive reuse of ISO definitions outside the licensed document is not permitted [27].

In the ontology development presented here, classes are therefore defined based on terms extracted from ISO terminology, with definitions either reused or rephrased where appropriate. Each class is explicitly linked to its ISO source via `rdfs:isDefinedBy` or `dc:source`, and the use of ISO content is strictly limited to short excerpts and definitional guidance within the ontology, with no substantial portions of the standard redistributed.

As an example, ISO 6892-1 (Metallic Materials — Tensile Testing, Part 1: Method of Test at Room Temperature) [28] has previously served as a source for developing tensile testing ontologies [26, 29, 30]. For this study, two new versions of a tensile testing ontology were developed based on the corresponding ISO standard. The first version was created by the human experts. While existing tensile testing ontologies developed by various groups served as a source of inspiration, those ontologies differ from the one presented here in that they import additional ontologies beyond BFO and draw terms and definitions from multiple external sources. The reference ontology developed here, by contrast, is

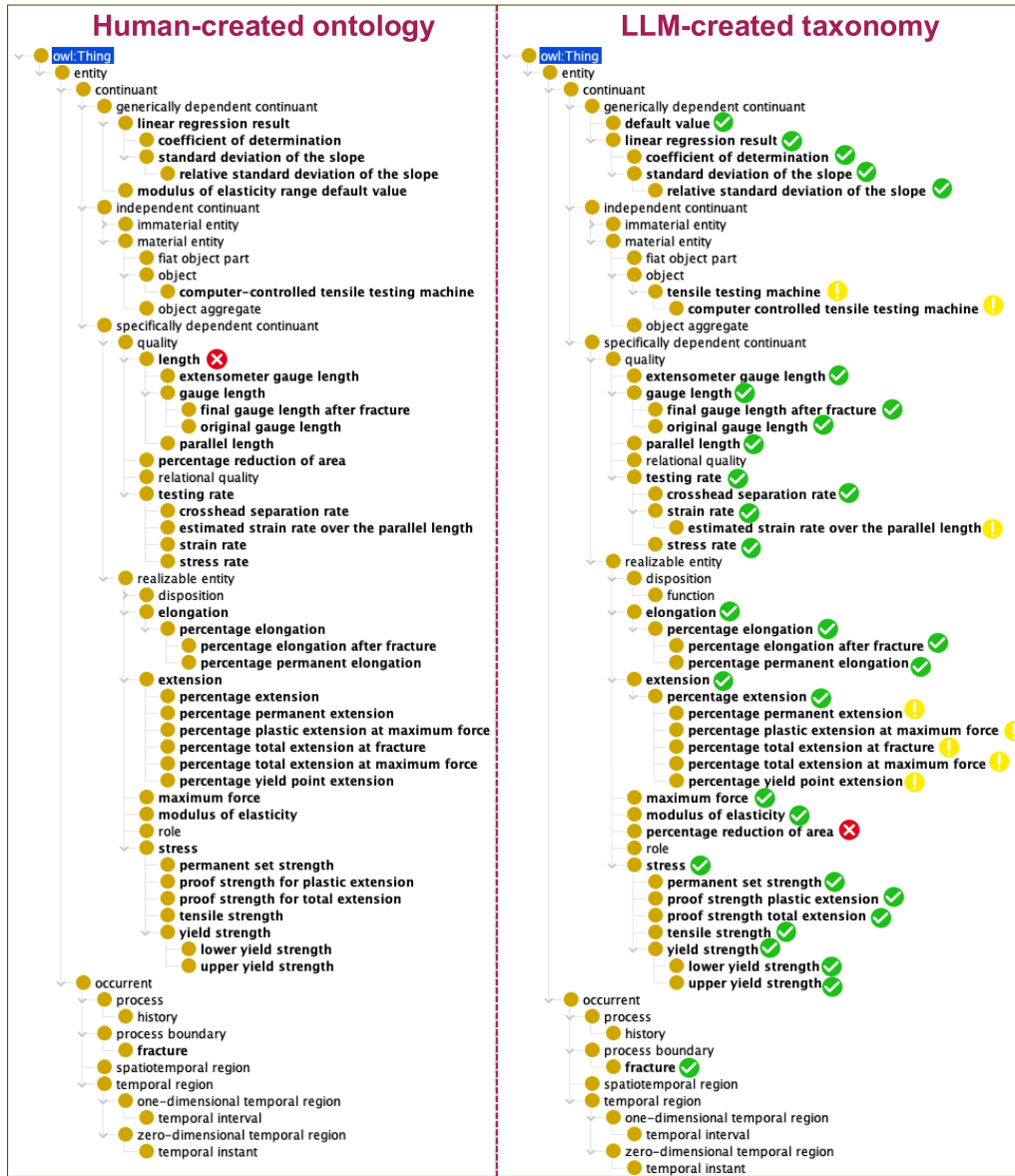


Figure 3: Class hierarchy comparison between the human- and LLM-created BFO-compliant tensile testing taxonomies derived from ISO 6892-1, visualized in Protégé.

strictly grounded in the ISO standard text.

The second taxonomy version was generated using LLM4BFO. For this purpose, the publicly available "Terms and Definitions" section of the standard was extracted from the ISO website [28] and provided as a PDF input to the tool. As described previously, the generation pipeline processes the terms and definitions from the input text and applies ontology engineering rules during class creation. In this regard, original definitions are paraphrased and reformulated into Aristotelian-style definitions, and class labels are adapted according to ontology labeling conventions. Fig. 3 presents the class hierarchies of both ontologies as visualized in Protégé.

Table 3 presents the quantitative evaluation results comparing the human-generated and LLM-generated taxonomies. Approximately 40 terms were extractable from the source text, and both the human and LLM4BFO produced ontologies covering these terms. Of the approximately 40 classes present in both ontologies, 39 were successfully matched, demonstrating that LLM4BFO extracts ontology classes from the source text in a manner highly consistent with a human expert.

The lexical label similarity achieved an F_1 -score of 0.950. The high similarity is largely attributable to the nature of the source document, i.e., the terms defined in ISO standards are already highly formalized,

Table 3

Quantitative evaluation of lexical label similarity, semantic definition similarity, and BFO hierarchy alignment between the human- and LLM-created tensile testing taxonomies derived from ISO 6892-1.

Matched classes (#)	Label lexical similarity (F_1)	Definitions semantic similarity (%)	Classes BFO hierarchy match (F_1)	Local parent hierarchy match (F_1)
39 (from 40)	0.950	76.85	0.961	0.823

expert-approved, and universally accepted, reducing the need for substantial label modifications by either the human expert or the LLM. Lexical differences appear only in a small number of cases involving highly specialized technical terms. For example, the class "default value" was represented differently across the two taxonomies; the human-created ontology used the more explicit and semantically refined label "modulus of elasticity range default value," whereas the LLM-generated taxonomy retained a less specific formulation.

Regarding class definitions, the ontology generation pipeline paraphrases and reformulates source definitions into Aristotelian-style definitions according to the ontology engineering rules implemented via LLM4BFO. When comparing the LLM-created definitions against the original source definitions, certain differences are therefore expected; these are primarily lexical and syntactic rather than fundamentally semantic, as reflected by a semantic similarity score of 76.85%. Compared to the egg boiling example, ISO standards provide a significantly more realistic and domain-relevant evaluation scenario, as they contain professionally curated terminology, formally written definitions, and highly specialized domain knowledge.

The color-coded indicators in Fig. 3 were used to compare the class hierarchies of both taxonomies. The results show that the hierarchical structures are highly similar, with most classes marked in green, indicating strong hierarchical agreement. Some yellow-labeled cases reflect local hierarchy differences. For example, the LLM introduced a class "tensile testing machine" under bfo:Object to organize subclasses such as "computer-controlled tensile testing machine," while the human expert introduced a superclass "length" to group related subclasses such as "gauge length" and "parallel length." Both cases reflect a shared tendency to generalize highly specific terms by introducing broader superclass structures.

Additional local hierarchy differences are also present. In the human-generated ontology, different "extension" classes were modeled as parallel sibling classes, whereas the LLM-generated ontology introduced a superclass "percentage extension" under which the remaining extension-related classes were organized hierarchically, a structure that is arguably more coherent from an ontology engineering perspective. A similar restructuring can be observed for the "strain rate" related classes.

The only major BFO hierarchy mismatch between the two ontologies concerns the class "percentage reduction of area." In the human-generated ontology, this class is modeled as a subclass of bfo:Quality, whereas in the LLM-generated ontology it is modeled as a subclass of bfo:RealizableEntity. This discrepancy reflects a known challenge in ontology engineering, as distinguishing between "qualities" and "realizable entities" can be non-trivial even for domain experts, a difficulty further evidenced by the fact that existing tensile testing ontologies in the literature assign this term to different hierarchical positions [26, 29, 30]. Based on the provided definition and the ontology engineering constraints implemented in the tool, LLM4BFO selected bfo:RealizableEntity as the appropriate parent class.

Overall, the hierarchies of the two ontologies remain highly similar. The BFO hierarchy matching F_1 -score of 0.961 confirms that LLM4BFO can successfully generate high-quality BFO-aligned ontology hierarchies. The local parent hierarchy similarity score of 82.3% reflects cases where either the human expert or the LLM introduced additional local superclasses (such as "percentage extension," "length," or "tensile testing machine") differences that are considered reasonable and acceptable in both ontology versions.

5. Conclusion

This paper introduces LLM4BFO, a pipeline for automated generation of BFO-compliant taxonomies from structured domain texts using LLMs. The pipeline integrates structured JSON input, embedded BFO reference grounding, constrained instruction prompting, and deterministic Turtle serialization to guide term and definition extraction from text, class hierarchy construction, and Aristotelian definition generation. Evaluation on two structured text sources, with results compared against human-created ontologies, showed that LLM4BFO performs well across all evaluated dimensions: high lexical label similarity (F_1 -score of ~ 0.96), meaningful semantic definition preservation ($\sim 77\%$), and reliable BFO hierarchy alignment (F_1 -score of ~ 0.96). These results indicate that LLM4BFO can serve as a reliable assistant for ontology engineers, substantially reducing the manual effort required in the early stages of BFO-compliant taxonomy development. Several limitations, however, remain. The current implementation does not support advanced modeling scenarios such as multiple inheritance and polyhierarchical structures. Expert validation also remains essential, as isolated cases of BFO misclassification and definition inconsistency were observed. Future work will address these limitations and extend evaluation to a broader range of domain texts and ontology engineering tasks.

Declaration on Generative AI

During the preparation of this work, the authors used GPT-5.5 to assist with grammar checking and rephrasing of the manuscript text. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of this publication.

References

- [1] A. S. Lippolis, et al., Ontology generation using large language models, in: E. Curry, et al. (Eds.), *The Semantic Web*, Springer Nature Switzerland, Cham, 2025, pp. 321–341. doi:10.1007/978-3-031-94575-5_18.
- [2] A. S. Lippolis, M. J. Saeedizade, R. Keskisarkka, A. Gangemi, E. Blomqvist, A. G. Nuzzolese, Assessing the capability of large language models for domain-specific ontology generation, 2025. doi:10.48550/arXiv.2504.17402. arXiv:arXiv:2504.17402.
- [3] P. Mateiu, A. Groza, Ontology engineering with large language models, in: *2023 25th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, 2023, pp. 226–229. doi:10.1109/SYNASC61333.2023.00038.
- [4] N. Fathallah, A. Das, S. D. Giorgis, A. Poltronieri, P. Haase, L. Kovriguina, NeOn-GPT: A large language model-powered pipeline for ontology learning, in: *The Semantic Web: ESWC 2024 Satellite Events*, Springer Nature Switzerland, Cham, 2025, pp. 36–50. doi:10.1007/978-3-031-78952-6_4.
- [5] N. Fathallah, S. Staab, A. Algergawy, LLMs4Life: Large language models for ontology learning in life sciences, 2024. doi:10.48550/arXiv.2412.02035. arXiv:arXiv:2412.02035.
- [6] S. Toro, et al., Dynamic retrieval augmented generation of ontologies using artificial intelligence (DRAGON-AI), *Journal of Biomedical Semantics* 15 (2024) 19. doi:10.1186/s13326-024-00320-3.
- [7] N. Matentzoglou, et al., MapperGPT: Large language models for linking and mapping entities, 2024.
- [8] Z. Liu, et al., OntoTune: Ontology-driven self-training for aligning large language models, 2025. doi:10.48550/arXiv.2502.05478. arXiv:arXiv:2502.05478.
- [9] D. Garijo, M. Poveda-Villalón, E. Amador-Domínguez, Z. Wang, R. García-Castro, O. Corcho, LLMs for ontology engineering: A landscape of tasks and benchmarking challenges, 2024.
- [10] S. Tsaneva, S. Vasic, M. Sabou, LLM-driven ontology evaluation: Verifying ontology restrictions with ChatGPT, 2024.

- [11] H. B. Giglou, J. D'Souza, F. Engel, S. Auer, LLMs4OM: Matching ontologies with large language models, in: *The Semantic Web: ESWC 2024 Satellite Events*, volume 15344 of *Lecture Notes in Computer Science*, Springer Nature Switzerland, Cham, 2025, pp. 25–35. doi:10.1007/978-3-031-78952-6_3.
- [12] C. Benson, A. Sculley, A. Liebers, J. Beverley, My ontologist: Evaluating bfo-based ai for definition support, in: *STIDS 2025: Semantic Technology for Intelligence, Defense, and Security*, 2025.
- [13] H. B. Nasrabadi, E. Norouzi, H. Sack, B. Skrotzki, Performance evaluation of upper-level ontologies in developing materials science ontologies and knowledge graphs, *Advanced Engineering Materials* 27 (2025) 2401534. doi:10.1002/adem.202401534.
- [14] J. N. Otte, J. Beverley, A. Ruttenberg, BFO: Basic formal ontology, *Applied Ontology* 17 (2022) 17–43. doi:10.3233/AO-220262.
- [15] E. H. Granger, Aristotle on genus and differentia, *Journal of the History of Philosophy* 22 (1984) 1–23.
- [16] T. T. Procko, O. Ochoa, T. Elvira, Automatic generation of BFO-compliant aristotelian definitions in OWL ontologies with GPT, in: *2023 Fifth International Conference on Transdisciplinary AI (TransAI)*, 2023, pp. 141–146. doi:10.1109/TransAI60598.2023.00042.
- [17] M. P. Joachimiak, et al., The artificial intelligence ontology: LLM-assisted construction of AI concept hierarchies, *Applied Ontology* 19 (2024) 408–418. doi:10.1177/15705838241304103.
- [18] F. H. Rodrigues, A. G. Lopes, N. O. dos Santos, L. F. Garcia, J. L. Carbonera, M. Abel, On the use of ChatGPT for classifying domain terms according to upper ontologies, in: *Advances in Conceptual Modeling*, Springer Nature Switzerland, Cham, 2023, pp. 249–258. doi:10.1007/978-3-031-47112-4_24.
- [19] A. Lopes, J. Carbonera, D. Schmidt, L. Garcia, F. Rodrigues, M. Abel, Using terms and informal definitions to classify domain entities into top-level ontology concepts: An approach based on language models, *Knowledge-Based Systems* 265 (2023) 110385. doi:10.1016/j.knosys.2023.110385.
- [20] A. Lopes, J. Carbonera, F. Rodrigues, L. Garcia, M. Abel, How to classify domain entities into top-level ontology concepts using large language models, *Applied Ontology* 19 (2024) 311–333. doi:10.3233/AO-240032.
- [21] A. Lopes, J. Carbonera, N. Santos, F. Rodrigues, L. Garcia, M. Abel, Cross-domain classification of domain entities into top-level ontology concepts using BERT: A study case on the BFO domain ontologies, in: *Proceedings of the 26th International Conference on Enterprise Information Systems*, SCITEPRESS, Angers, France, 2024, pp. 141–148. doi:10.5220/0012557600003690.
- [22] A. Lopes, J. L. Carbonera, F. Rodrigues, L. F. Garcia, M. Abel, Unveiling influential factors in classifying domain entities into top-level ontology concepts: an analysis using GO and ChEBI ontologies, 2024.
- [23] H. B. Nasrabadi, F. Mahdavi, J. Waitelonis, H. Sack, LLM4BFO: Large language models for creating BFO-compliant taxonomies, 2026.
- [24] H. Beygi Nasrabadi, F. Mahdavi, LLM4BFO: Large Language Models for Building BFO-Aligned Ontologies, 2026. URL: <https://github.com/ISE-FIZKarlsruhe/LLM4BFO>.
- [25] <https://github.com/bfo-ontology/BFO-2020>, 2026. Accessed: 2026-06-01.
- [26] H. B. Nasrabadi, T. Hanke, M. Weber, M. Eisenbart, F. Bauer, R. Meissner, G. Dziwis, L. Tikana, Y. Chen, B. Skrotzki, Toward a digital materials mechanical testing lab, *Computers in Industry* 153 (2023) 104016.
- [27] <https://www.iso.org/copyright.html>, 2026. Accessed: 2026-06-01.
- [28] <https://www.iso.org/obp/ui/en/#iso:std:iso:6892:-1:ed-3:v1:en>, 2026. Accessed: 2026-06-01.
- [29] M. Schilling, B. Bayerlein, P. von Hartrott, J. Waitelonis, H. Birkholz, P. Dolabella Portella, B. Skrotzki, Fair and structured data: A domain ontology aligned with standard-compliant tensile testing, *Advanced Engineering Materials* 27 (2025) 2400138.
- [30] C. E. DE NORMALISATION, E. K. F. NORMUNG, A guide to the development and use of standards compliant data formats for engineering materials test data (2010).