Exploring a Large Language Model for Transforming Taxonomic Data into OWL: Lessons Learned and Implications for Ontology Development

Filipi Miranda Soares[a,b,c,*], Antonio Mauro Saraiva[a,b], Luís Ferreira Pires[c], Luiz Olavo Bonino da Silva Santos[c,d], Dilvan de Abreu Moreira[e,b], Fernando Elias Corrêa[f,b], Kelly Rosa Braghetto[g,b], Debora Pignatari Drucker[h,b], Alexandre Cláudio Botazzo Delbem[e,b]

[a] *University of Sao Paulo, Polytechnic School, Computer Engineering and Digital Systems, Sao Paulo, 05508-010, Brazil*
[b] *University of Sao Paulo, Center for Artificial Intelligence (C4AI), Sao Paulo, 05508-020, Brazil*
[c] *University of Twente, Faculty of Electrical Engineering, Mathematics and Computer Science, Semantics, Cybersecurity & Services Group, Enschede, 7522 NB, Netherlands*
[d] *Leiden University Medical Center, Human Genetics, Leiden, 2333 ZC, Netherlands*
[e] *University of Sao Paulo, Institute of Mathematics and Computer Science, Sao Carlos, 13566-590, Brazil*
[f] *University of Sao Paulo, Luiz de Queiroz College of Agriculture, Center for Advanced Studies on Applied Economics, Piracicaba, 13400-970, Brazil*
[g] *University of Sao Paulo, Institute of Mathematics and Statistics, Sao Paulo, 05508-090, Brazil*
[h] *Embrapa Digital Agriculture, Campinas, 13083-886, Brazil*

Abstract

Managing scientific names in ontologies that represent species taxonomies is challenging due to the ever-evolving nature of these taxonomies. Manually maintaining these names becomes increasingly difficult when dealing with thousands of scientific names. To address this issue, this paper investigates the use of ChatGPT-4 to automate the development of the `Organism` module in the Agricultural Product Types Ontology (APTO) for species classification. Our methodology involved leveraging ChatGPT-4 to extract data from the GBIF Backbone API and generate OWL files for further integration in APTO. Two alternative approaches were explored: (1) issuing a series of prompts for ChatGPT-4 to execute tasks via the BrowserOP plugin and (2) directing ChatGPT-4 to design a Python algorithm to perform analogous tasks. Both approaches rely on a prompting method where we provide instructions, context, input data, and an output indicator. The first approach showed scalability limitations, while the second approach used the Python algorithm to overcome these challenges, but it struggled with typographical errors in data handling. This study highlights the potential of Large language models like ChatGPT-4 to streamline the management of species names in ontologies. Despite certain limitations, these tools offer promising advancements in automating taxonomy-related tasks and improving the efficiency of ontology development.

Keywords: ChatGPT; artificial intelligence; knowledge graph; agriculture; taxonomy.

---

[*] Corresponding author: Filipi Miranda Soares (f.mirandasoares@utwente.nl; ORCID: 0000-0002-0674-7960).

1. Introduction

Ontologies capture a well-defined and well-expressed shared conceptualization of a domain [1]. Ontologies essentially specify the kinds of entities relevant to a particular domain and the relationships among such entities, thereby constructing a large semantic structure that represents the key aspects of domain knowledge [2]. Ontologies also play a crucial role in facilitating the integration of heterogeneous data sources, enabling interoperability among information systems [3].

Ontology development is a comprehensive process involving the conceptualization step, during which domain experts and ontologists work together to specify significant concepts and relationships peculiar to a particular domain. Such collaboration guarantees that the established ontology sufficiently addresses the intricacies of the domain while being responsive to new knowledge and uses [4, 5]. Ontologies facilitate the structured representation of knowledge, enhancing interoperability and reusability in modeling real-world concepts [6].

Ontologies in the life sciences domain often incorporate taxonomic classifications to represent and organize knowledge about organisms. Biologists define taxonomy as “the theory and practice of identifying, describing, naming, and classifying organisms” [7]. A taxonomy can be considered an ontology, as it formalizes hierarchical relationships and categories among organisms. However, the science of taxonomy faces numerous challenges from the dynamic nature of scientific discovery, as scientists continuously find new species and revise earlier established classifications [8]. The current debate over competing classification systems, such as the Linnaean hierarchy versus phylogenetic taxonomy, further compounds this situation by not reaching a consensus [9]. Additionally, maintaining consistency in naming conventions across languages and disciplines, managing synonyms (multiple names for the same organism), and resolving homonyms (the same name referring to different organisms) are also issues that have to be handled [10].

Despite these challenges, taxonomic classifications are a useful resource that can be utilized within ontologies for the classification of organisms. Current approaches to integrating taxonomic data into ontologies are, however, based predominantly on manual curation or semi-automated techniques that involve considerable domain knowledge and computational resources. While prior studies have explored the role of Large Language Models (LLMs) in Knowledge Graph (KG) generation and ontology development, most focus on unstructured text extraction, semantic alignment, or conceptual modeling rather than direct automation of taxonomic data transformation.

This study systematically evaluates LLMs for converting taxonomic data into OWL, leveraging ChatGPT-4 to extract data from the Global Biodiversity Information Facility (GBIF) Backbone API and automate ontology generation. Unlike previous works focused on knowledge extraction, this research explores two approaches – direct API querying via ChatGPT and a Python-based method guided by LLM-generated code – to assess scalability, reliability, and integration into a domain-specific ontology. Specifically, it investigates a GPT-based approach for integrating taxonomic data into the Agricultural Product Types Ontology (APTO), enhancing its accuracy and resolving ambiguities in agricultural product names, which often vary across languages, regions, and cultural contexts.

The paper is organized as follows: Section 2 reviews related work, while Section 3 introduces APTO. Section 4 outlines the methods employed in this study. Sections 5 and 6 present the results of Approach 1 and 2, respectively. Section 7 explains the integration of species classification into APTO, and Section 8 concludes with the final considerations.

2. Related Work

The application of LLMs in KG and ontology development has gained attention due to their capability to process and generate conceptual schemas from natural language text sources [11]. Recent studies have demonstrated that LLMs, such as ChatGPT, can significantly support tasks in ontology engineering, including concept extraction, class hierarchy generation, and semantic relationship identification [11, 12].

One of the most common uses of LLMs in ontology development is automating concept extraction and populating ontologies with domain knowledge. For example, Aggarwal et al. [13] experimented with using LLMs in academic ontology development in engineering fields, with promising results in extracting and structuring domain knowledge. Zaitoun et al. [14] also employed LLMs to enrich biomedical ontologies, demonstrating that they are able to identify missing concepts and relations. Trajanoska et al. [15] and Reis et al. [16] demonstrated how LLMs can be utilized for enhancing KG construction from unstructured text.

LLMs also facilitate semantic relationship discovery and ontology alignment. Amini et al. [17] investigated LLMs for complex ontology alignment, using contextual embeddings to improve relationship detection. Matentzoglu et al. [18] demonstrated how LLMs like MapperGPT enhance entity linking and mapping precision. Wu et al. [19] suggested an approach using online clustering with LLMs for ontology expansion.

Additionally, studies have explored applying LLMs for entity resolution, focusing on identifying, matching, and reconciling entities across different data sources. Hees [20] examined LLMs for aligning non-ontological data to ontologies, focusing on semantic interoperability in the energy sector. Using GPT-3.5 and GPT-4, the study aligned datasets from the ENERSHARE project with the Semantic TreeHouse ontology framework. The findings suggest that while LLMs aid entity resolution, they function best as recommendation tools rather than deterministic alignment methods.

Zhang et al. [21] explored weakly-supervised taxonomy enrichment, demonstrating how LLMs can identify discriminative features to enhance hierarchical structures. Their study introduced self-supervised entity resolution techniques and seed-guided taxonomy construction, showing that minimal human supervision enables LLMs to build structured ontologies from text corpora.

Hofer et al. [22] analyzed LLM-driven entity reconciliation in KG construction, emphasizing the need for incremental ontology updates. Their findings indicate that while LLMs generate entity alignment candidates, a hybrid approach – combining LLMs with rule-based validation – is necessary for accuracy. The study highlights LLM applications in ontology alignment, particularly in biomedical KGs, where they refine taxonomic structures while maintaining semantic consistency.

Chen et al. [23] examined schema-based KGs for taxonomic reconciliation, leveraging OWL ontologies, RDFS, and SHACL. The study underscores the importance of ontology mapping repositories (e.g., UMLS Metathesaurus, BioPortal, MONDO) in resolving cross-domain taxonomic inconsistencies. Additionally, it explores LLM integration in entity resolution, particularly in instance matching and automated ontology extension.

These works emphasize the potential for LLMs to automate and improve ontology engineering, which is a fundamental concern investigated in this study. Table 1 summarizes how these works relate to our study.

Despite these developments, integrating LLMs into ontology engineering is not challenge-free. Challenges related to model interpretability and reliability [12, 24, 25], domain specificity,

Table 1: Relationship between related studies and this paper

Study	**Focus area**	**Relationship with this study**
Aggarwal et al. [13]	LLMs for scholarly ontology generation	Automated concept and relationship extractions in ontology development
Zaitoun et al. [14]	LLMs for biomedical ontology augmentation	Demonstrates LLMs' ability to identify missing ontological relationships
Trajanoska et al. [15] & Reis et al. [16]	LLMs for KG construction from unstructured text	LLM automation of ontology engineering tasks
Amini et al. [17]	LLMs for ontology alignment using contextual embeddings	Enhances taxonomic data alignment through contextual embeddings
Matentzoglu et al. [18]	LLMs for entity linking and mapping	Improves entity matching, crucial for taxonomic reconciliation
Wu et al. [19]	LLMs for ontology expansion using online clustering	Provides insights into LLM-based ontology expansion
Hees [20]	LLMs for semantic interoperability and ontology alignment	Applies LLMs to ontology alignment, relevant for aligning non-ontological data to an ontology
Zhang et al. [21]	LLMs for weakly-supervised taxonomy enrichment	Supports taxonomy enrichment with minimal human supervision
Hofer et al. [22]	LLMs for incremental ontology updates and entity reconciliation	Enhancing the scalability of ontology completion, updates, and creation using LLMs to extract and integrate new concepts from structured data, such as tabular datasets.
Chen et al. [23]	KGs for taxonomic reconciliation and LLM integration	Demonstrates how schema-based KGs aid taxonomic integration with LLMs

and computational costs [26] are some of the key obstacles. Furthermore, the reliability of LLM-generated outputs in highly technical fields remains a concern, as hallucination and data bias continue to affect the consistency and accuracy of results [27]. Overcoming these challenges is crucial to guaranteeing the robustness and real-world applicability of LLM-supported ontology engineering.

3. The Agricultural Product Types Ontology (APTO)

APTO is a domain-specific ontology developed to standardize terminology for agricultural product types within the Brazilian commodities market and comprises 3,333 statements. It incorporates over 200 product types identified in datasets from three Brazilian agricultural price index agencies, namely Cepea[1], Ipea[2], and Conab[3].

The ontology adopts a modular approach, comprising two core segments: the Organism module and the Product Type module. The Organism module links agricultural products to their biological origins using taxonomic classifications aligned with the GBIF Backbone Taxonomy. In contrast, the Product Type module categorizes products based on their origin (e.g., plant-based, animal-based, or inorganic) and processing state (raw, processed, or by-product). This duality allows APTO to capture the intricate relationships between products, their sources, and their processing stages, making it generally suitable for applications ranging from semantic web integration to agricultural market analysis.

We reused existing domain ontologies such as AGROVOC[4] [28] and Agrotermos[5] [29] when designing APTO. AGROVOC is a multilingual thesaurus developed by the Food and Agriculture

---

[1] URL: `https://www.cepea.esalq.usp.br/en`

[2] URL: `https://www.ipeadata.gov.br/`

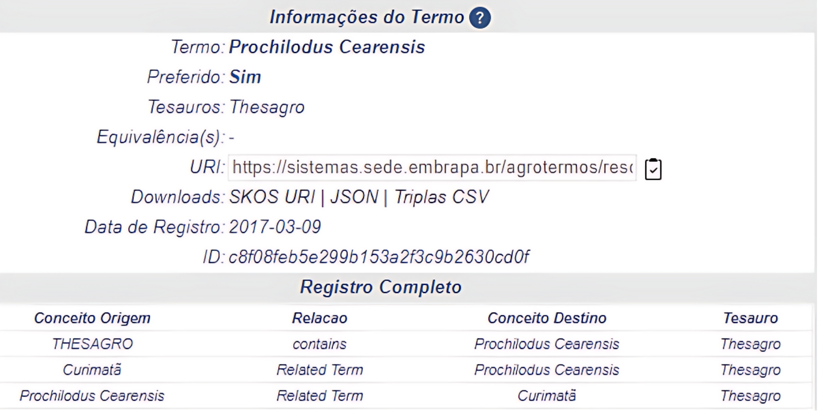
[3] URL: `https://www.conab.gov.br/info-agro/precos`

[4] URL: `https://agrovoc.fao.org/browse/agrovoc/en/`

[5] URL: `https://sistemas.sede.embrapa.br/agrotermos/`

Organization (FAO) to standardize terminology in agriculture, fisheries, forestry, and related domains [28, 30]. In contrast, Agrotermos is a Brazilian agricultural vocabulary designed to support research and dissemination of knowledge specific to the Brazilian agricultural sector [31]. Additionally, Agrotermos serves as a terminology provider to AGROVOC for Brazilian Portuguese and incorporates concepts from AGROVOC into its vocabulary, ensuring interoperability and alignment between the two resources.

Despite their utility, a review of these vocabularies revealed gaps, particularly in the consistency and currency of taxonomic names. The dynamic nature of taxonomies, where scientific names are frequently updated due to synonymization and other taxonomic revisions, presents a significant challenge to maintaining ontological accuracy. For example, Agrotermos contains outdated taxonomic names due to the lack of automated mechanisms for verifying authoritative taxonomic sources for synonymization. Similarly, AGROVOC relies on a manual process for incorporating scientific names into its vocabulary as specified by its editorial guidelines [32], which can result in similar issues. Fig. 1 illustrates this problem with the species *Prochilodus cearensis* Steindachner, 1911, which is a preferred term in Agrotermos, yet it is declared a synonym of *Prochilodus brevis* Steindachner, 1875 in the GBIF Backbone Taxonomy [33].



Informações do Termo

Termo: **Prochilodus Cearensis**
Preferido: **Sim**
Tesauros: Thesagro
Equivalência(s): -
URI: https://sistemas.sede.embrapa.br/agrotermos/res(
Downloads: SKOS URI | JSON | Triplas CSV
Data de Registro: 2017-03-09
ID: c8f08feb5e299b153a2f3c9b2630cd0f

Registro Completo

Conceito Origem	Relacao	Conceito Destino	Tesauro
THESAGRO	contains	Prochilodus Cearensis	Thesagro
Curimatã	Related Term	Prochilodus Cearensis	Thesagro
Prochilodus Cearensis	Related Term	Curimatã	Thesagro

Figure 1: Example of synonym in Agrotermos in Portuguese[6].

We did not import species name classes directly from Agrotermos and AGROVOC to prevent these inconsistencies from propagating into APTO. Instead, we developed a method to extract up-to-date data from the GBIF Backbone Taxonomy and convert it into OWL format for seamless integration into the ontology.

The development of APTO followed the SABiO methodology for ontology development [34], a framework encompassing five main phases: (1) Purpose Identification and Requirements Elicitation, (2) Ontology Capture and Formalization, (3) Design, (4) Implementation, and (5) Testing. This paper focuses on part of the Design phase, where we explored the use of ChatGPT to streamline the integration of taxonomic data into the ontology.

APTO has been implemented in OWL and is accessible through endpoints, including a

---

[6] URL: `https://sistemas.sede.embrapa.br/agrotermos/resources/c8f08feb5e299b153a2f3c9b2630cd0f`

SPARQL endpoint[7], a text-search interface[8], and an OWL/XML serialization[9]. For Findability, Accessibility, Interoperability, and Reusability (FAIR) purposes, the ontology has been made available on AgroPortal[10].

4. Methods

Ontologies and KGs are closely related, with ontologies often serving as foundational frameworks for creating and interpreting KGs. Ontologies provide structured representations of domain-specific knowledge, including classes, relationships, and constraints. This logical foundation aligns with the principles of KGs, which are graph-based structures used to model interconnected data for reasoning and discovery [35, 36]. Ontologies not only aid in formalizing the semantics of KGs but also extend their capabilities by enabling advanced reasoning tasks [36]. Furthermore, they enhance KG construction by ensuring semantic interoperability and consistency, particularly when generating KGs from textual data [37]. The overlap between ontologies and KGs illustrates how these paradigms coalesce to support complex data representation and reasoning [38]. Given this close relationship, ontology development can benefit from recent techniques employed in KG construction, particularly applying LLMs.

The method employed in this case builds upon the framework outlined by [39] for LLM-augmented KG development to explore this synergy. The authors address different techniques using LLMs in KG development, including LLM-augmented KG Construction, and describe two approaches: end-to-end KG construction and distilling KGs from LLMs. According to [39], we can understand these approaches as follows:

- **End-to-end KG construction** is an innovative approach utilizing LLMs to streamline KG creation from raw text, bypassing the conventional multi-stage process [39]. Studies like [40] have developed methods integrating LLMs for named entity recognition and relation extraction, employing dual BERT-based classifiers to identify relationships and their orientations between entities [39]. This method allows for the direct assembly of KGs, showcasing models like Grapher and PiVE that facilitate KG generation from textual descriptions or correct errors in existing KGs using iterative verification with smaller LLMs [39].

- **Distilling KGs** from LLMs capitalizes on the inherent knowledge encoded within LLMs to extract and structure this information into KGs [39]. Techniques like COMET [41] leverage LLMs to generate commonsense KGs by training on seed sets of knowledge tuples, enabling the LLM to produce novel, high-quality knowledge tuples [39]. This approach underscores the potential of LLMs as tools for enhancing KG construction and as vast repositories of implicit knowledge ripe for extraction and organization into explicitly structured forms [39].

The methodological approach employed in this paper is similar to the End-to-End KG Construction but less complex since we do not deal with extraction from raw text. The data input

---

[7] URL: `https://data.pldn.nl/FilipiSoares/APTO/sparql`

[8] URL: `https://data.pldn.nl/FilipiSoares/APTO/elasticsearch/APTO-Text-search/text`

[9] URL: `https://w3id.org/APTO#`

[10] URL: `https://agroportal.lirmm.fr/ontologies/APTO`

we used in the ontology construction is structured as JSON files with representative and precise names, which makes it easier to process (and more precise) than natural language input.

In this research, we used ChatGPT-4 to improve the quality of an ontology. Two alternative approaches were explored: (1) issuing a series of prompts for ChatGPT-4 to execute tasks via the BrowserOP plugin and (2) directing ChatGPT-4 to design a Python algorithm to perform analogous tasks. We followed the workflow activities shown in Fig. 2 in both approaches. The following sections discuss each activity in detail.

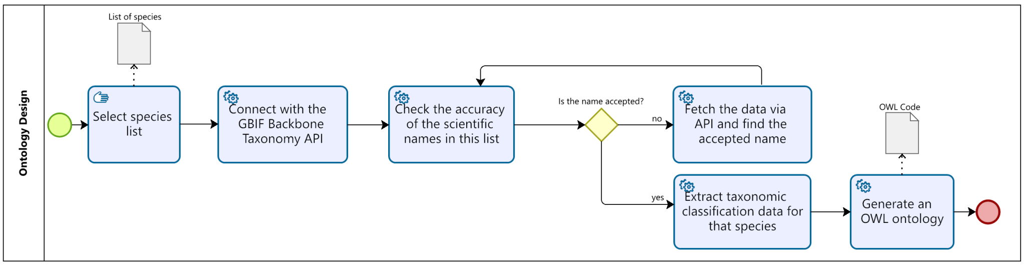


Figure 2: Workflow for the taxonomic data transformation.

4.1. *Selecting the Species List*

We started by selecting a list of plant and animal species related to all agricultural product types in APTO. This step involved mapping product types from three key Brazilian datasets onto commodity price indexes, namely the agricultural prices index from the Center for Advanced Studies on Applied Economics (Cepea) [42], the IpeaData Macroeconomic Indicators [43], and the 'Companhia Nacional de Abastecimento' (Conab) agricultural prices database [44]. We based the association between product types and species on resources such as Agrotermos, Agrovoc, and the scientific literature, producing two lists: an Animal species list with 14 scientific names and a Plant species list with 74 names, both lists with names of species or genus level.

Another criterion for selecting these species was their agricultural relevance, particularly within Brazilian agronomic and livestock domains. The list includes economically significant species, widely cultivated, and well-documented in key Brazilian agricultural data repositories. Given Brazil's diverse agricultural landscapes, these species provide a representative yet incomplete view of the taxonomic groups involved in commercial agriculture.

While the selection is representative of Brazilian agrobiodiversity, it has limitations when generalizing to global taxonomies. Many agricultural taxonomies include species from other regions with different climatic, economic, and ecological contexts. As this study focuses on Brazilian datasets, certain species with regional importance elsewhere may be underrepresented. However, the approach developed in this study is not limited to Brazilian species. It can be expanded to include species from other regions, as long as they are represented in the GBIF Backbone Taxonomy.

4.2. *Connecting with the GBIF Backbone Taxonomy API*

GBIF is an international network and data infrastructure supported by governments worldwide. Its primary goal is to provide open access to biodiversity data for all types of life on

Earth, accessible from anywhere [45]. Recognized as the most extensive biodiversity data infrastructure regarding the volume of observations [46], GBIF serves many stakeholders, including researchers, policymakers, and other decision-makers. It achieves this by aggregating data from numerous sources, standardizing the information, and making it freely available for public use [47].

For the Organism module in APTO, we imported scientific names from the GBIF Backbone Taxonomy [48]. GBIF provides an API[11] for extracting taxonomic classifications. This API operates on data stored within the GBIF Checklist Bank, which indexes every registered checklist dataset across the GBIF network [49]. To consume these RESTful JSON web services, GBIF internally utilizes a Java web service client [49]. Using ChatGPT-4 and the BrowserOp plugin, we automated the retrieval of taxonomic data for each species via the GBIF API.

The GBIF Backbone Taxonomy is a comprehensive synthetic classification system that underpins GBIF's integration of diverse name-based information from various sources, including occurrence datasets, species pages, and external databases like the Encyclopedia of Life (EOL), Genbank, and the International Union for Conservation of Nature and Natural Resources (IUCN) red list database, facilitating consistent taxonomic searches and cross-referencing across resources [48]. Regular updates expand its scope by merging additional scientific names from authoritative datasets into the Catalogue of Life-based structure [50], enhancing coverage beyond family-level classifications [48]. The taxonomy also incorporates identifiers for Operational Taxonomic Units (OTUs) from barcoding projects like iBOL and UNITE, employing consensus-based methods to connect Barcode Index Numbers (BINs) and Species Hypotheses (SHs) to taxon names and classifications [48]. Available for download in various formats, the GBIF Backbone Taxonomy data represents a critical resource for taxonomic research, drawing from over 105 sources to ensure broad and accurate name coverage [48].

4.3. Checking the Accuracy of Scientific Names

International nomenclature codes govern the assignment of scientific names to species, establishing principles and rules to ensure clarity and consistency across the scientific community. This process involves the formal description and publication of each species, adhering to specific criteria set by these codes. Over time, scientific names may become deprecated due to various factors, such as the discovery of prior valid names overlooked in earlier assessments or new phylogenetic insights that necessitate reclassification [51, 52]. As a result, the nomenclatural landscape is dynamic, with taxonomists revising and updating names to reflect the most accurate understanding of biological relationships. These updates lead to synonymizing many species names [51, 52].

We reviewed each name on the species list to determine whether it was accepted or a synonym. In Approach 1, we used ChatGPT to check the name status through the GBIF API. In Approach 2, we implemented this process using the Python Taxonomy Converter.

4.4. Extracting Multi-level Taxonomic Classification Data

Species classification follows various conventions, leading to multi-level taxonomies that organize life forms into hierarchical categories [53]. In this study, we adopted a seven-tiered framework for the taxonomic classification of species, consisting of kingdom, phylum, class, order, family, genus, and species, aligned with established biological taxonomy principles [53].

---

[11] URL: `https://api.gbif.org/v1/`

This hierarchical structure is crucial for understanding the relationships and distinctions among different life forms [53]. During the classification phase, we tasked both applications — ChatGPT integrated with the BrowserOp plugin and the Python Taxonomy Converter — with extracting these seven levels of classification for each species listed or six levels for names at the genus level.

4.5. Transforming Taxonomic Data Into OWL

Finally, we used the extracted taxonomic information to generate OWL/XML files, incorporating hierarchical relationships that accurately reflect the underlying species taxonomy upon which they were based.

5. Approach 1: ChatGPT with BrowserOp Plugin

5.1. Prompt Specification and Results

The process consisted of incorporating a series of instructions into the ChatGPT 4 model, as shown in Table 2. This study's prompt engineering methodology builds on Giray's work [54], which categorizes the components of an effective prompt into four main elements:

- **Instruction**: A directive that specifies the task, guiding the model's behavior and ensuring it aligns with the desired outcome.
- **Context**: Relevant background information or external details provide additional knowledge, enabling the model to generate accurate and context-aware responses.
- **Input data**: The core query or data to be processed serves as the primary driver of the model's understanding and response generation.
- **Output indicator**: An indicator of the expected response format, such as a brief answer, detailed explanation, or structured output, shaping the model's solution delivery.

We built the prompt based on these four elements, as shown in Table 2.

Table 2: Prompt Specification

Element	**Prompt**
Instruction	Build an ontology for species names.
Context	Access the GBIF API [using the BrowserOp plugin]: `https://api.gbif.org/v1/`
Input data	Extract the full taxonomic classification for each name in this list, including kingdom, phylum, class, order, family, genus, and species. In some cases, the name on the list is at the species level, and in other cases, at the genus level.
Input data	If a name has the status ‘deprecated‘, look for the accepted name for that species.
Output indicator	This taxonomy should be organized as OWL classes and subclasses (do not create individuals or instances). Each class should have a URI from GBIF and an RDF label with the name for that class, with the language tag (considering that scientific names are written in Latin, include the language tag “lat”). Finally, generate an OWL code, including all the necessary namespaces.

The species names used as **Input data** are listed in Table 3. The first species in this list, *Apis mellifera*, served as a test case where the ChatGPT model utilized the BrowserOp plugin to send a request to the GBIF API. This plugin fetched real-time data from the web [55], and facilitated the interaction as detailed in Listing 1. As specified in **Context**, the call to the GBIF API performed with the BrowserOP plugin successfully retrieved the desired data. In our directive to ChatGPT, we specified a connection with the GBIF API's base URL (`https://api.gbif.org/v1/`), which hosts various endpoints catering to different data dimensions, including but not limited to the Registry API, Species API, Occurrence API, and Literature API, each serving distinct purposes such as registry lookup, species matching, occurrence data retrieval, and literature references, respectively. Despite the lack of explicit instruction on which endpoint to use, ChatGPT identified and targeted the appropriate API endpoint for species matching, which is `https://api.gbif.org/v1/species/`.

Table 3: List of animal scientific names

Taxon rank	Taxon name
Species	*Apis mellifera*
Species	*Bos taurus*
Subspecies	*Capra aegagrus hircus*
Species	*Ovis aries*
Genus	*Sus*
Species	*Bombyx mori*
Species	*Prochilodus cearensis*
Species	*Prochilodus scrofa*
Species	*Prochilodus margravii*
Species	*Semaprochilodus insignis*
Species	*Semaprochilodus taeniunes*
Species	*Colossoma mitrei*
Species	*Arapauma gigas*
Genus	*Tilapia*

Listing 1: Response from the GBIF API to the GPT-BrowserOP call

```
// Request to BrowserOP
{
  "link": "https://api.gbif.org/v1/species/match?name=Apis%20mellifera"
}

// Response from BrowserOp
{
  "result": {
    "content": "{\"usageKey\":1341976,\"scientificName\":\"Apis mellifera
        Linnaeus, 1758\",\"canonicalName\":\"Apis mellifera\",\"rank\":\"
        SPECIES\",\"status\":\"ACCEPTED\",\"confidence\":99,\"matchType\":
        \"EXACT\",\"kingdom\":\"Animalia\",\"phylum\":\"Arthropoda\",\"
        order\":\"Hymenoptera\",\"family\":\"Apidae\",\"genus\":\"Apis\",\"
        species\":\"Apis mellifera\",\"kingdomKey\":1,\"phylumKey\":54,\"
        classKey\":216,\"orderKey\":1457,\"familyKey\":4334,\"genusKey\":13
        34757,\"speciesKey\":1341976,\"synonym\":false,\"class\":\"Insecta
        \"}\n",
    "rules": [
      "Use the language that the user previously used or the language
          requested by the user.",
      "Based on the provided data (and all historical retrieved data if the
           results need more than 1 requests to be completely provided),
          respond to the user's request. No need to include other
          irrelevant words in your answer.",
      "Always extract url or source from provided content as a support for
          your answer.",
      "Ensure the information provided is accurate, clear, and concise.
          Avoid overly complex or lengthy explanations when a simple,
          direct response will suffice, while in the meantime, make sure to
           provide all the information the user will find valuable based on
           your correct understanding of user's intention.",
      "If a conclusive answer cannot be generated based on the provided
          content, respond to user's query first and then attempt to
```

```
          interpret the user's intent and suggest ways for them to refine their query for a more successful result.",
      "If the provided data includes authoritative source such as [wiki] or [official website], organize the response based primarily on data from these sources.",
      "If the user's query contains a URL, give a brief description of the site and summarize it based on the content of the page."
    ]
  },
  "code": 0
}
```

The API response shown in Listing 1 adheres to the prompt specification (refer to Table 2), demonstrating the successful extraction of detailed taxonomic classification data for the species *Apis mellifera*. As specified in the prompt's **Input data**, ChatGPT retrieved the complete taxonomic hierarchy, including the ranks: kingdom, phylum, class, order, family, genus, and species. Listing 1 confirms the accurate extraction of all ranks, formatted as key-value pairs (e.g., `kingdom:Animalia`).

The prompt further specified that the ontology generated from the retrieved (**Output indicator** in Table 2) data should incorporate GBIF URIs for each class representing a taxon. Listing 1 shows that each taxon is assigned a unique `key` corresponding to its taxonomic rank (e.g., `kingdomKey:1`, `genusKey:133475`). GBIF assigns a unique numeric ID to each entity in its Backbone Taxonomy, forming the URI that uniquely identifies every scientific name in its catalog. For example, the response indicates Animalia is linked to the `kingdomKey` value 1, which is represented by the GBIF URI `https://www.gbif.org/species/1`. These URIs ensure that each taxon is unambiguously identified within the ontology.

The prompt also required ChatGPT to verify the species' name's status. The API response in Listing 1 shows the name status as `ACCEPTED`, indicating that the species name is valid and recognized within the taxonomy catalog. This outcome confirms ChatGPT's successful execution of this verification step.

Moreover, Listing 1 shows the `rules` block, which outlines a set of operational guidelines not explicitly defined in the prompt, suggesting they originate from ChatGPT's model internal rules to enhance the response's comprehensiveness and relevance.

Finally, the prompt in Table 2 defined that the output should be an OWL file generated from the data extracted via the API. Initially, GPT-4 created OWL code using `https://api.gbif.org/v1/species/` as the namespace for the ontology classes. This URI is a reference point within the GBIF API, linking to JSON data files describing scientific names and their relationships, as illustrated in Listing 2.

Listing 2: JSON data for the Animalia class (https://api.gbif.org/v1/species/1)

```
{
  "key": 1,
  "nubKey": 1,
  "nameKey": 130188353,
  "taxonID": "gbif:1",
  "sourceTaxonKey": 1,
  "kingdom": "Animalia",
  "kingdomKey": 1,
  "datasetKey": "d7dddbf4-2cf0-4f39-9b2a-bb099caae36c",
  "constituentKey": "d7dddbf4-2cf0-4f39-9b2a-bb099caae36c",
  "scientificName": "Animalia",
  "canonicalName": "Animalia",
```

```
  "vernacularName": "Animals",
  "authorship": "",
  "nameType": "SCIENTIFIC",
  "rank": "KINGDOM",
  "origin": "SOURCE",
  "taxonomicStatus": "ACCEPTED",
  "nomenclaturalStatus": [],
  "remarks": "",
  "numDescendants": 2981931,
  "lastCrawled": "2023-08-22T23:20:59.545+00:00",
  "lastInterpreted": "2023-08-22T22:11:51.237+00:00",
  "issues": []
}
```

However, after careful consideration, we decided to use URIs from the `https://www.gbif.org/species/` endpoint instead. We chose this endpoint because it provides access to valuable enriched resources for end users, such as taxon photos, occurrence statistics, and related information. Although we did not import these supplementary resources into the ontology, users can access them by clicking on the URL, aligning with linked data principles.

To apply this change, we directed ChatGPT to regenerate the OWL code using `https://www.gbif.org/species/` as the namespace. Listing 3 shows the updated output.

Listing 3: GPT-generated OWL Representation for Taxonomic Hierarchy

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
         xmlns:owl="http://www.w3.org/2002/07/owl#">

    <!-- Kingdom Animalia -->
    <owl:Class rdf:about="https://www.gbif.org/species/1">
        <rdfs:label xml:lang="lat">Animalia</rdfs:label>
    </owl:Class>

    <!-- Phylum Arthropoda -->
    <owl:Class rdf:about="https://www.gbif.org/species/54">
        <rdfs:label xml:lang="lat">Arthropoda</rdfs:label>
        <rdfs:subClassOf rdf:resource="https://www.gbif.org/species/1"/>
    </owl:Class>

    <!-- Class Insecta -->
    <owl:Class rdf:about="https://www.gbif.org/species/216">
        <rdfs:label xml:lang="lat">Insecta</rdfs:label>
        <rdfs:subClassOf rdf:resource="https://www.gbif.org/species/54"/>
    </owl:Class>

    <!-- Order Hymenoptera -->
    <owl:Class rdf:about="https://www.gbif.org/species/1457">
        <rdfs:label xml:lang="lat">Hymenoptera</rdfs:label>
        <rdfs:subClassOf rdf:resource="https://www.gbif.org/species/216"/>
    </owl:Class>

    <!-- Family Apidae -->
    <owl:Class rdf:about="https://www.gbif.org/species/4334">
        <rdfs:label xml:lang="lat">Apidae</rdfs:label>
        <rdfs:subClassOf rdf:resource="https://www.gbif.org/species/1457"/>
    </owl:Class>

    <!-- Genus Apis -->
```

```
    <owl:Class rdf:about="https://www.gbif.org/species/1334757">
        <rdfs:label xml:lang="lat">Apis</rdfs:label>
        <rdfs:subClassOf rdf:resource="https://www.gbif.org/species/4334"/>
    </owl:Class>

    <!-- Species Apis mellifera -->
    <owl:Class rdf:about="https://www.gbif.org/species/1341976">
        <rdfs:label xml:lang="lat">Apis mellifera</rdfs:label>
        <rdfs:subClassOf rdf:resource="https://www.gbif.org/species/1334757
            "/>
    </owl:Class>

</rdf:RDF>
```

We repeated this process for all taxa listed in Table 3. ChatGPT confirmed the taxon names *Bos taurus*, *Ovis aries*, Genus *Sus*, *Semaprochilodus insignis*, *Semaprochilodus taeniurus*, *Arapaima gigas*, and *Tilapia* as accepted names within the GBIF Backbone Taxonomy. Then, it successfully generated the corresponding OWL code for these taxa. The OWL files are available on Zenodo [1].

ChatGPT identified the remaining taxa as having outdated or deprecated names, classifying them as synonyms, as shown in Table 4. These taxon names were listed as preferred terms in the Agrotermos thesaurus but were absent from AGROVOC, the two vocabularies used as the foundation for building this ontology. An exception is *Capra hircus*, which we did not find in either of these vocabularies.

Table 4: Synonymous scientific names and their corresponding accepted names identified by ChatGPT

Target name	**Synonym of**
Capra hircus	*Capra aegagrus*
Prochilodus cearensis	*Prochilodus brevis*
Prochilodus scrofa	*Prochilodus lineatus*
Prochilodus margravii	*Prochilodus argenteus*
Colossoma mitrei	*Piaractus mesopotamicus*

Agrotermos and AGROVOC include concepts for representing “goat,” but with limitations. In Agrotermos, the term “Cabra” (Portuguese for goat) is not explicitly associated with its scientific species. In AGROVOC, *Capra hircus* is modeled as an entry term for this concept. To address this gap, we referred to the classification provided on Wikipedia, which defines “goat” species as *Capra hircus*. We added this name to the list, which turned out to be outdated.

ChatGPT retrieved the corresponding accepted name from the GBIF Backbone Taxonomy for each synonym identified and generated OWL code incorporating these validated names, as shown in Listing 4.

Listing 4: Output After Resolving Synonyms

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
         xmlns:owl="http://www.w3.org/2002/07/owl#">

                                 ...

    <!-- Species Prochilodus brevis (for Prochilodus cearensis) -->
    <owl:Class rdf:about="https://www.gbif.org/species/2352151">
        <rdfs:label xml:lang="lat">Prochilodus brevis</rdfs:label>
        <rdfs:subClassOf rdf:resource="https://www.gbif.org/species/2352148
            "/>
```

```
    </owl:Class>

    <!-- Species Prochilodus lineatus (for Prochilodus scrofa) -->
    <owl:Class rdf:about="https://www.gbif.org/species/2352154">
        <rdfs:label xml:lang="lat">Prochilodus lineatus</rdfs:label>
        <rdfs:subClassOf rdf:resource="https://www.gbif.org/species/2352148
            "/>
    </owl:Class>

    <!-- Species Prochilodus argenteus (for Prochilodus margravii) -->
    <owl:Class rdf:about="https://www.gbif.org/species/2352177">
        <rdfs:label xml:lang="lat">Prochilodus argenteus</rdfs:label>
        <rdfs:subClassOf rdf:resource="https://www.gbif.org/species/2352148
            "/>
    </owl:Class>

</rdf:RDF>
```

In Listing 4, we tasked ChatGPT with generating OWL code for three species within the same genus: “Prochilodus Cearensis,” “Prochilodus Scrofa,” and “Prochilodus Margravii.” However, during our first attempt to extract data from the GBIF API using these names, ChatGPT reported that no data was available for these taxa. We discovered that this issue stemmed from formatting errors in the species names. According to international standards for species nomenclature [56, 57], species names should consist of a capitalized genus name followed by a lowercase specific epithet, often including the naming authority and year of identification. The names we used in the initial query, copied directly from the Agrotermos vocabulary, had capitalization errors in the specific epithets (e.g., “Cearensis” instead of “cearensis”), which rendered them unrecognizable to the GBIF API.

After correcting the formatting to align with these standards, we reissued the query, and ChatGPT successfully retrieved the taxonomic data. Table 5 provides a detailed log of this interaction, highlighting the issue and the resolution process.

Following the initial incident, we intentionally repeated the same mistake by submitting the species name “Colossoma Mitrei” with the same capitalization error in the specific epithet. ChatGPT corrected the error autonomously before sending the request to the GBIF API, demonstrating its ability to learn iteratively from prior interactions.

This adaptability reflects ChatGPT’s advanced machine-learning techniques, allowing it beyond simple input processing to refine its comprehension and response strategies over time. Such functionality exemplifies a core characteristic of Artificial Intelligence (AI): the ability to internalize lessons from previous interactions and apply this knowledge to similar situations in the future, thereby reducing errors and enhancing operational efficiency [58, 59].

In ontology development, this AI-driven error correction potentially reduces manual effort and enhances data quality. Studies [58, 59, 60, 61] highlight the growing role of AI in fostering more intelligent and efficient system interactions, making it a transformative tool in fields that demand adaptability and precision.

5.2. *Limitations*

This approach presented limitations when handling extensive species lists. For this reason, we did not move forward with generating the OWL code for the plant species listed in the ontology since the list of plant species is larger than the animal species list.

Table 5: Iteration with ChatGPT

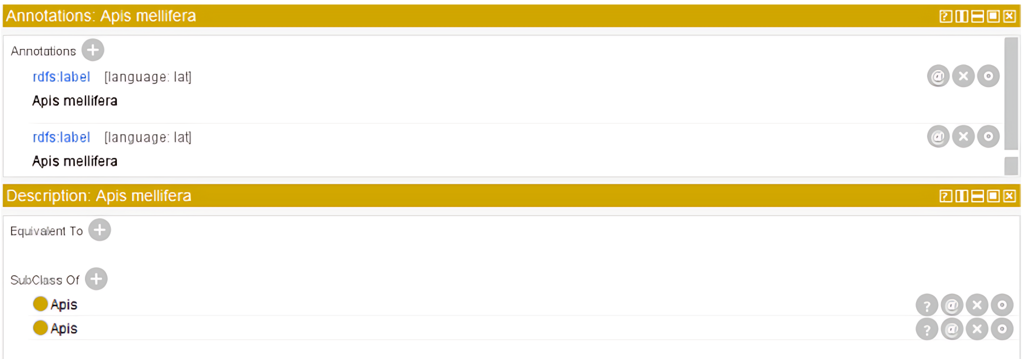
Agent	Prompt
👤	Do it [repeat the steps described in Table 2] for these three species: Prochilodus Cearensis, Prochilodus Scrofa, and Prochilodus Margravii.
֍	The attempts to match the species names "Prochilodus Cearensis," "Prochilodus Scrofa," and "Prochilodus Margravii" with the GBIF database did not yield any results due to too little confidence in the matches. This means that the specific names provided might not be directly available in the GBIF database under those exact spellings, or are not recognized at the species level in the database. For taxonomic research and ontology building, accurate species names are crucial. It is possible that these names might have synonyms under which they are listed in databases, or there might be slight variations in spelling or taxonomy that affect the search results. In such cases, further steps could include: Verifying the scientific names for spelling or taxonomic updates. Searching for broader taxonomic groups (e.g., the genus "Prochilodus") and then narrowing down to the species level based on additional sources or databases. Consulting specialized taxonomic databases or literature for the most current classification and synonyms. If you have alternative or broader taxonomic categories in mind or need assistance with another query, please let me know how I can assist further.
👤	Try again with Prochilodus cearensis, Prochilodus scrofa, and Prochilodus margravii.
֍	Based on the data retrieved from the GBIF API for the species Prochilodus cearensis, Prochilodus scrofa, and Prochilodus margravii, here's the ontology in OWL format. It's important to note that all three names are recognized as synonyms for other species within the GBIF database: [provided the OWL code shown in Listing 4. After the code block, it commented:] This OWL code represents the full taxonomic classification for the species Prochilodus cearensis (synonym for Prochilodus brevis), Prochilodus scrofa (synonym for Prochilodus lineatus), and Prochilodus margravii (synonym for Prochilodus argenteus), including the higher taxon levels (kingdom, phylum, class, order, family, genus) with each class having a valid URI from GBIF and a rdf label of the name for that class in Latin.

Moreover, the processing time increases with the list's size, impacting this method's scalability. For example, the processing time to call the GBIF API and generate the ontology for the species *Apis mellifera* was 37 seconds. For the three species *Prochilodus cearensis*, *Prochilodus scrofa*, and *Prochilodus margravi*, which were synonyms, the processing time was 2 minutes and 5 seconds. This approach proved efficient for smaller clusters of species names, particularly when the names are taxonomically related (within the same genus or family). However, we observed that reliability decreases for lists with more than five names, where the likelihood of hallucinations increases. Initially, we attempted to process the entire list of species names from Table 3 through ChatGPT in a single batch. However, this approach resulted in multiple inaccuracies, including the generation of URIs non-existent in the GBIF database, incorrect association between valid URIs and labels, and occasional omissions of tasks, such as adding RDF labels. We opted for a more controlled approach to address these issues by submitting names to ChatGPT individually or in smaller groups (at most three species each time). This strategy significantly improved the accuracy of the results.

Another challenge was integrating multiple OWL files using Protégé, which resulted in duplicated labels and parent-child relationships as illustrated in Fig. 3. Despite attempts to consolidate these files using Protégé's merge function, these redundancies persisted, but this can be also considered as a limitation of Protégé for handling these multiple files rather than a problem with ChatGPT-generated codes.

Another limitation of this approach is the challenge of reproducibility due to ongoing updates in ChatGPT's technology. During the initial experiments, ChatGPT plugins such as BrowserOP were still operational. However, as of March 2024, OpenAI replaced these plugins with GPTs, and ChatGPT no longer has direct access to the internet.

To assess the feasibility of reproducing the results, we attempted to replicate the experiments on March 6, 2025, using three different GPT models: BrowserOP (the former plugin), ChatGPT-4o, and Data Analyst. In all three cases, the models returned the same message indi-


Annotations: Apis mellifera

Annotations

rdfs:label [language: lat]
Apis mellifera

rdfs:label [language: lat]
Apis mellifera

Description: Apis mellifera

Equivalent To

SubClass Of
● Apis
● Apis

Figure 3: Duplicated labels and relationships in the ontology visualization in Protégé.

cating their inability to make direct API calls, confirming that ChatGPT currently lacks internet access. Therefore, this approach is no longer viable unless ChatGPT regains internet access in the future.

To evaluate whether a similar approach could work with alternative AI models, we tested the same prompt using DeepSeek V3 and Google Gemini 2.0. However, the results were highly inaccurate and unreliable. Although both models generated API responses that structurally resembled the expected format (similar to Listing 1), the responses contained incorrect taxon keys that did not match the expected labels. Additionally, neither model correctly identified that the species in this test, *Prochilodus scrofa*, was a synonym.

Even after multiple refinements to the prompt, including providing more details about API syntax and sample responses, both models continued to generate hallucinated results that did not accurately reflect the expected outputs. These issues indicate that using this approach with DeepSeek V3 and Gemini 2.0 would require further fine-tuning or prompt engineering to mitigate these errors. The full prompts and interaction transcripts with DeepSeek and Google Gemini are available on Zenodo 1.

In conclusion, this approach is currently not reproducible with ChatGPT due to the lack of internet access. Additionally, attempts to apply this methodology to other AI models, such as DeepSeek V3 and Gemini 2.0, require further adjustments to reduce hallucinations and improve reliability. Therefore, we recommend using Approach 2 as a more viable solution for generating an OWL ontology for species taxonomy.

6. Approach 2: Python Algorithm for Converting GBIF Taxonomic Data to OWL

Building on the steps outlined in Table 2 and the prompt interactions described in Section 5, we tasked ChatGPT with developing a Python algorithm to generate OWL files from data retrieved via the GBIF Backbone Taxonomy API. However, the initial version of the algorithm introduced redundancies in the generated OWL code by repeating higher taxonomic levels for each taxon. For example, the class *Animalia* was redefined for every taxon describing an animal species rather than being defined once and used as a superclass for the relevant species. To address this issue, we provided ChatGPT with the revised instruction outlined in Table 6.

The algorithm second version fixed redundant taxonomic level definitions but introduced a new issue: improper class-subclass relationships in the OWL code. When reported to ChatGPT,

Table 6: Iteration with ChatGPT: fixing repetitions

Agent	Prompt
👤	It should present the superior taxon levels once in the code when they are the same for multiple species. Animalia, for instance, is a taxon common to all species in this list.
֎	To address the issue of repeating higher taxon levels for multiple species, you can modify the script to accumulate unique taxon levels across all species before generating the OWL representation. This approach ensures that each taxon level (e.g., kingdom, phylum) is represented only once, regardless of how many species share it.

it responded as shown in Table 7.

Table 7: Iteration with ChatGPT: fixing relationships

Agent	Prompt
֎	“To ensure the OWL output correctly represents the `subClassOf` relationships and avoids duplicating common higher taxon levels across species, let’s refine the approach. We’ll adjust the script to build a hierarchy that maintains these relationships and ensures each taxon level is included only once.”

In this third trial, we got an error: “AttributeError: ‘list’ object has no attribute ‘replace.’ A typo in the list of names caused this error. We tasked ChatGPT to review the species list, and it detected “*Semaprochilodus taeniunes*” as a typo, suggesting the correct name as “*Semaprochilodus taeniurus*.” ChatGPT inference was correct since this is also the accepted name in the GBIF Backbone Taxonomy. After reviewing this name and reentering the list, the algorithm worked as expected.

Finally, we tasked ChatGPT with generating the documentation for the algorithm, which we named GBIF Taxonomy OWLizer. The complete algorithm and its documentation are available on Zenodo [2].

6.1. Script Overview

This Python script follows a pipeline to fetch taxonomic classification data for a species list from the GBIF API and generates an OWL representation. It can handle multiple species names, representing higher taxonomic levels only once, even if they are shared across species, avoiding duplication of classes as reported in the limitations of Approach 1. It also checks synonyms and looks for the accepted names. The script is written in Python 3.x and requires the `requests` library.

Functions

This script met the requirements specified in Table 2, including the following functions:

- `fetch_gbif_data(scientific_name)`: Fetches taxonomic classification from the GBIF API for a given scientific name.
- `fetch_synonyms(species_key)`: Fetches synonyms for a given species key from the GBIF API. It checks the accepted scientific name of a species, ensuring that the OWL ontology only includes the correct names.
- `accumulate_taxa(data, taxa)`: Accumulates unique taxa and their relationships from the GBIF data.
- `validate_accepted_name(data, taxa)`: Ensures that only the accepted scientific name is included in the taxa dictionary by validating against synonyms.
- `generate_owl(taxa)`: Generates an OWL/XML representation from the accumulated taxa.

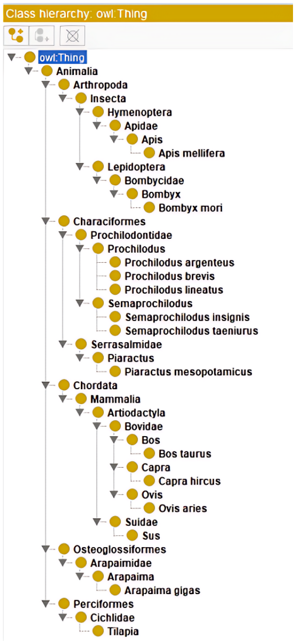
Listing 5 shows an example of usage for the list of names from Table 3. The output was an OWL/XML file (`Animalia.xml` on Zenodo [1]). Subsequent inspection of this file in Protégé revealed a well-structured class hierarchy, depicted in Fig. 4. This visualization confirmed the accuracy of the hierarchical relationships and the appropriate formatting of the generated OWL code.

Listing 5: Example of usage

```python
species_names = [
    "Apis mellifera", "Bos taurus", "Capra aegagrus hircus", "Ovis aries", "Sus", "Bombyx mori", "Prochilodus cearensis", "Prochilodus scrofa", "Prochilodus margravii", "Semaprochilodus insignis", "Semaprochilodus taeniurus", "Colossoma mitrei", "Arapaima gigas", "Tilapia"
]

taxa = {}
for name in species_names:
    data = fetch_gbif_data(name)
    if data:
        accumulate_taxa(data, taxa)
        validate_accepted_name(data, taxa)

owl_data = generate_owl(taxa)
print(owl_data)
```



```
owl:Thing
  Animalia
    Arthropoda
      Insecta
        Hymenoptera
          Apidae
            Apis
              Apis mellifera
        Lepidoptera
          Bombycidae
            Bombyx
              Bombyx mori
    Characiformes
      Prochilodontidae
        Prochilodus
          Prochilodus argenteus
          Prochilodus brevis
          Prochilodus lineatus
        Semaprochilodus
          Semaprochilodus insignis
          Semaprochilodus taeniurus
      Serrasalmidae
        Piaractus
          Piaractus mesopotamicus
    Chordata
      Mammalia
        Artiodactyla
          Bovidae
            Bos
              Bos taurus
            Capra
              Capra hircus
            Ovis
              Ovis aries
          Suidae
            Sus
    Osteoglossiformes
      Arapaimidae
        Arapaima
          Arapaima gigas
    Perciformes
      Cichlidae
        Tilapia
```

Figure 4: Visualization of classes hierarchy in Protégé.

Testing the Algorithm with a Larger List of Species Names

The scalability limitation of Approach 1 prevented the generation of OWL codes for long species lists. In contrast, the second approach works significantly faster. To evaluate its efficiency, we conducted a test with 74 plant species listed in the file `PlantSpeciesList.py` available on Zenodo [1]. As an initial step, we asked ChatGPT to verify the spelling of the species names in this list using only its internal knowledge, without connecting to the GBIF API.

Although we did not explicitly request ChatGPT to identify synonyms – since our algorithm handles this functionality – it flagged *Euterpe edulis* as a synonym of *Euterpe oleracea*. However, subsequent verification using authoritative sources, including the GBIF Backbone Taxonomy [48] and the Plants of the World Online catalog [62], confirmed that both *Euterpe edulis* and *Euterpe oleracea* are valid and accepted species names. This misidentification highlights the potential for inaccuracies when ChatGPT's analysis is based solely on its internal knowledge rather than being anchored to reliable external sources for validating taxonomic names.

Despite the incorrect synonym suggestion, we continued with the task. The algorithm performed efficiently, generating OWL code for nearly all the listed species – except for *Triticum x Secale* – in just 2 minutes and 31 seconds. By comparison, the first approach required 2 minutes and 5 seconds to generate OWL code for only three species. This stark contrast highlights the superior scalability of the second approach, which successfully processed 74 species at a similar time. The complete OWL code can be found as `plantae.xml` on Zenodo [1].

The exclusion of *Triticum x Secale* from the output was due to its hybrid name syntax, which the GBIF API does not recognize. This probably happens because hybridization occurs at the individual level, and not at the species level, as highlighted by [63].

Names of hybrids are structured to indicate their origin from two different species or genera. The × indicates hybrid status in botanical nomenclature. It is placed between the genus name and the specific epithet for interspecific hybrids or between two genera for intergeneric hybrids [64]. For instance, the intergeneric hybrid between the genera Triticum (wheat) and Secale (rye), is denoted as "Triticum × Secale," where the "×" explicitly marks its hybrid nature.

In zoology, hybrids are also identified using similar conventions, although they are less standardized and may vary depending on the taxonomic authority. Hybrids are often represented by combining the parent species' names or genera, sometimes without a specific symbol [56].

Hybrid names can have synonyms or be reclassified over time as more is learned about their genetics and relationships to parent species. Managing these changes and ensuring that the ontology reflects the most current understanding can be challenging. Representing the genetic and taxonomic relationships between hybrids and their parent species would require the ontology to support complex hierarchical and networked structures. This complexity can pose challenges for both the design of the ontology and for users querying the ontology to understand the relationships between different entities.

6.2. *Web Application with User Interface*

Based on the Python script of Approach 2, we developed the GBIF Taxonomy OWLizer[12], which is a web-based application that allows users to convert species names into an OWL taxonomy using the GBIF API. It provides a simple interface for users to enter species names, fetch their taxonomic classification, and download the generated OWL file. This tool is designed for biologists, taxonomists, and ontology researchers needing structured taxonomy data in OWL format.

---

[12] URL: `https://filipi-soares.github.io/TaxonOWLConverter/`

Functionalities:

- Users can input the names of one or multiple species.
- The app queries the GBIF API to fetch taxonomic classifications.
- The app converts the taxonomy data into an OWL ontology format.
- Users can copy and paste or download the generated OWL taxonomy file.
- Previously searched species are cached in local storage, allowing instant retrieval without re-fetching data.

Technologies used:

- HTML, CSS (Bootstrap): Provides a simple and responsive user interface.
- JavaScript: Handles API requests, caching, and file downloads.
- jQuery: Simplifies DOM manipulation and AJAX requests.
- Flask (Python Back-end on Render): Processes API requests and fetches taxonomic data from GBIF.
- GBIF API: Provides taxonomic classification data.
- GitHub Pages: Hosts the front-end (static website) for free.
- Render (Free Tier): Hosts the back-end Flask API that fetches data.
- UptimeRobot: Sends periodic requests to prevent the back-end from going to sleep.

6.3. Limitations

Approach 2 successfully addressed the scalability limitations of Approach 1. However, this algorithm remains sensitive to typographical errors in the input data. To mitigate this, we recommend a preliminary step in which users utilize ChatGPT to review the list of species names and correct typos. This workflow combines the strengths of both tools: ChatGPT's ability to verify and correct spellings and the algorithm's efficiency in processing species names and performing synonym checks.

While the GBIF Taxonomy OWLizer web-based app provides a functional and accessible way to generate OWL representations of taxonomic data, it has certain limitations due to free-tier services. The back-end, hosted on Render's free tier, enters a sleep state after 15 minutes of inactivity, causing delays in API responses when it needs to restart. To mitigate this, UptimeRobot periodically pings the back-end, keeping it active most of the time. However, this workaround is not entirely reliable and may still result in occasional downtime. Since GitHub Pages is used to host the front-end, the application can only serve static content, requiring an external back-end to handle API requests.

Furthermore, the app caches responses using local storage, meaning previously fetched data is only available on the same device and browser. This limits persistent caching across multiple sessions or devices, which could be improved by integrating a more robust storage solution such as IndexedDB or a cloud-based database. Despite these limitations, the current implementation remains an effective and free-to-use tool for taxonomists and ontology researchers.

7. Integrating Species Classification in APTO

After generating and integrating the OWL file with species classifications into APTO Organism module, each product type was linked to its corresponding species. This association is essential for accurately representing the meaning of commercial product names, particularly within the regional context where they are traded. Establishing these precise connections ensures semantic clarity and helps prevent errors arising from ambiguous or incorrect associations.

APTO offers several examples that underscore the importance of this type of modeling. For example, a frequent ontological modeling error appears in sources such as Agrotermos, Agrovoc, and even in Wikipedia, concerning the term “Pimenta.” In Brazilian Portuguese, ‘Pimenta’ is often translated into English as “pepper.” While this translation might seem correct, it oversimplifies the term and creates semantic confusion. In Brazilian Portuguese, ‘Pimenta’ refers to the reddish or yellowish (sometimes green when collected unripe) berries from plants of the genus *Capsicum*. Examples include ‘Pimenta malagueta’ (*Capsicum frutescens*), ‘Pimenta cumari’ (*Capsicum baccatum var. praetermissum*), and ‘Pimenta dedo-de-moça’ (*Capsicum baccatum*).

In resources such as Agrovoc [65] and Agrotermos [66], the concept of ‘Pimenta’ is translated into English as “pepper” and associated with the species *Piper nigrum*. In Brazil, however, *Piper nigrum* refers explicitly to the plant that produces black pepper, known locally as ‘Pimenta-do-reino.’

In Agrotermos, *Capsicum* sp. is listed as a related concept to ‘Pimenta,’ but there is also a relationship indicating that ‘Pimenta’ is produced by *Piper nigrum*. This misrepresentation creates confusion, as the term ‘Pimenta’ in Brazil is predominantly associated with *Capsicum* sp., while ‘Pimenta-do-reino’ refers to *Piper nigrum* (black pepper), as previously mentioned.

This issue is not limited to these two vocabularies. On Wikipedia, a search for ‘Pimenta’ in Portuguese retrieves an article describing ‘Pimentas’ derived from *Capsicum* sp., as shown in Fig. 5a. However, switching the article’s language to English leads to an article about “pepper,” which refers to *Piper nigrum*, as illustrated in Fig. 5b.

Despite being a common mistake, the organisms represented by these concepts are biologically distinct. Table 8 shows the biological classification of both species, highlighting their differences. *Capsicum* sp. and *Piper nigrum* are only related at the class level (*Magnoliopsida*), underscoring their distinct evolutionary and biological characteristics.

Table 8: Comparison of the classification between *Piper nigrum* and *Capsicum* sp., based on the GBIF Backbone Taxonomy [48].

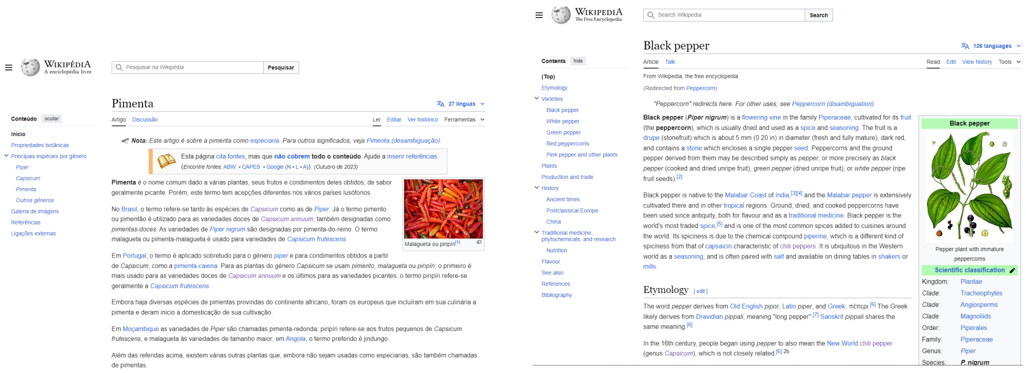
Taxon rank	**Piper nigrum**	**Capsicum sp.**
Kingdom	Plantae	Plantae
Phylum	Tracheophyta	Tracheophyta
Class	Magnoliopsida	Magnoliopsida
Order	Piperales	Solanales
Family	Piperaceae	Solanaceae
Genus	*Piper* L.	*Capsicum* L.
Species	*Piper nigrum* L.	—

The accurate English equivalent of the Brazilian concept “Pimenta” is “chili pepper.” However, in Agrovoc [67] and Agrotermos [68], the concept of “chili pepper” is incorrectly matched with the Portuguese term ‘Condimento picante’, which translates to “spicy condiment” in English. This translation conflates two distinct concepts from different hierarchical levels, treating them as language equivalents when they are not.

In Wikipedia, the “chili pepper” article translates into Portuguese as ‘Pimentas chili’. This translation is problematic because the term ‘Pimentas chili’ is not used in Brazil, leading to further confusion. These examples highlight how a single incorrect association can trigger a series of ontological inconsistencies across different terminology systems and languages.

To address this issue in APTO, we created two distinct classes, each explicitly linked to the correct biological classification:

- `Pepper: 'member of taxon' exactly 1` *`Piper nigrum`*
- `Chili pepper: 'member of taxon' some` *`Capsicum`*



Pimenta

Nota: Este artigo é sobre a pimenta como especiaria. Para outros significados, veja Pimenta (desambiguação).

Esta página cita fontes, mas que **não cobrem todo o conteúdo**. Ajude a inserir referências
(*Encontre fontes:* ABW • CAPES • Google (N • L • A)). *(Outubro de 2023)*

Pimenta é o nome comum dado a várias plantas, seus frutos e condimentos deles obtidos, de sabor geralmente picante. Porém, este termo tem acepções diferentes nos vários países lusófonos.

Malagueta ou piripíri[1]

No Brasil, o termo refere-se tanto às espécies de *Capsicum* como as de *Piper*. Já o termo pimento ou pimentão é utilizado para as variedades doces de *Capsicum annuum*, também designadas como *pimentas-doces*. As variedades de *Piper nigrum* são designadas por pimenta-do-reino. O termo malagueta ou pimenta-malagueta é usado para variedades de *Capsicum frutescens*.

Em Portugal, o termo é aplicado sobretudo para o género *piper* e para condimentos obtidos a partir de *Capsicum*, como a pimenta-caiena. Para as plantas do género *Capsicum* se usam *pimento*, *malagueta* ou *piripíri;* o primeiro é mais usado para as variedades doces de *Capsicum annuum* e os últimos para as variedades picantes; o termo piripíri refere-se geralmente a *Capsicum frutescens*.

Embora haja diversas espécies de pimentas provindas do continente africano, foram os europeus que incluíram em sua culinária a pimenta e deram inicio a domesticação de sua cultivação.

Em Moçambique as variedades de *Piper* são chamadas pimenta-redonda; piripíri refere-se aos frutos pequenos de *Capsicum frutescens*, e malagueta às variedades de tamanho maior; em Angola, o termo preferido é jindungo.

Além das referidas acima, existem várias outras plantas que, embora não sejam usadas como especiarias, são também chamadas de pimentas.

A Pimenta síria é uma mistura de especiarias que inclui as pimentas do reino e da jamaica.

(a) Pimenta on Wikipedia in Portuguese.

Black pepper

From Wikipedia, the free encyclopedia

(Redirected from Peppercorn)

"Peppercorn" redirects here. For other uses, see Peppercorn (disambiguation).

Black pepper (***Piper nigrum***) is a flowering vine in the family Piperaceae, cultivated for its fruit (the **peppercorn**), which is usually dried and used as a spice and seasoning. The fruit is a drupe (stonefruit) which is about 5 mm (0.20 in) in diameter (fresh and fully mature), dark red, and contains a stone which encloses a single pepper seed. Peppercorns and the ground pepper derived from them may be described simply as *pepper*, or more precisely as *black pepper* (cooked and dried unripe fruit), *green pepper* (dried unripe fruit), or *white pepper* (ripe fruit seeds).[2]

Black pepper is native to the Malabar Coast of India,[3][4] and the Malabar pepper is extensively cultivated there and in other tropical regions. Ground, dried, and cooked peppercorns have been used since antiquity, both for flavour and as a traditional medicine. Black pepper is the world's most traded spice,[5] and is one of the most common spices added to cuisines around the world. Its spiciness is due to the chemical compound piperine, which is a different kind of spiciness from that of capsaicin characteristic of chili peppers. It is ubiquitous in the Western world as a seasoning, and is often paired with salt and available on dining tables in shakers or mills.

Etymology [edit]

The word *pepper* derives from Old English *pipor*, Latin *piper*, and Greek: πέπερι.[6] The Greek likely derives from Dravidian *pippali*, meaning "long pepper".[7] Sanskrit *pippali* shares the same meaning.[6]

In the 16th century, people began using *pepper* to also mean the New World chili pepper (genus *Capsicum*), which is not closely related.[6]:2b

Black pepper

Pepper plant with immature peppercorns

Scientific classification

Kingdom:	Plantae
Clade:	Tracheophytes
Clade:	Angiosperms
Clade:	Magnoliids
Order:	Piperales
Family:	Piperaceae
Genus:	*Piper*
Species:	***P. nigrum***

(b) Black pepper as direct translation of 'Pimenta' on Wikipedia in English.

Figure 5: Comparison of 'Pimenta' and its translation as Black pepper in Wikipedia[13].

Multilingual vocabularies often face ontological challenges due to the complexity of modeling concepts across diverse regions and languages. However, such issues are not exclusive to global contexts; regionalisms can also create significant ontological challenges at the local level. A prominent example is `apto:Cheiro_verde`, a commonly used culinary herb mixture in Brazil. The term, which translates to "green smell" in English, is a staple in Brazilian cuisine, enhancing the flavor of soups, stews, and meats.

The composition of 'Cheiro verde' varies significantly across Brazil's regions, as illustrated in Fig. 6. In the Southeast, South, and Midwest, it typically consists of parsley and chives. In the Northeast, parsley is replaced by coriander; in the North, the mixture expands to include chives, coriander, and fitweed. This regional variation is particularly important for traditional Amazonian dishes such as 'Pato no tucupi' (duck in tucupi) and 'Tacacá' [69, 70].

In APTO, the variations of this concept have been modeled as shown in Listing 6. This approach allows even non-Portuguese speakers to understand the basic semantics of the concept despite its lack of a direct English translation. The class `Cheiro_verde` is defined as a combination of `Coriander and Chives`, `Parsley and Chives`, or `Coriander and Chives and Fitweed`, with each ingredient linked to its corresponding species. These combinations should ideally be associated with the specific Brazilian regions to enhance this modeling further.

---

[13] Wikipedia sources: `https://pt.wikipedia.org/wiki/Pimenta`, `https://en.wikipedia.org/wiki/Black_pepper`.

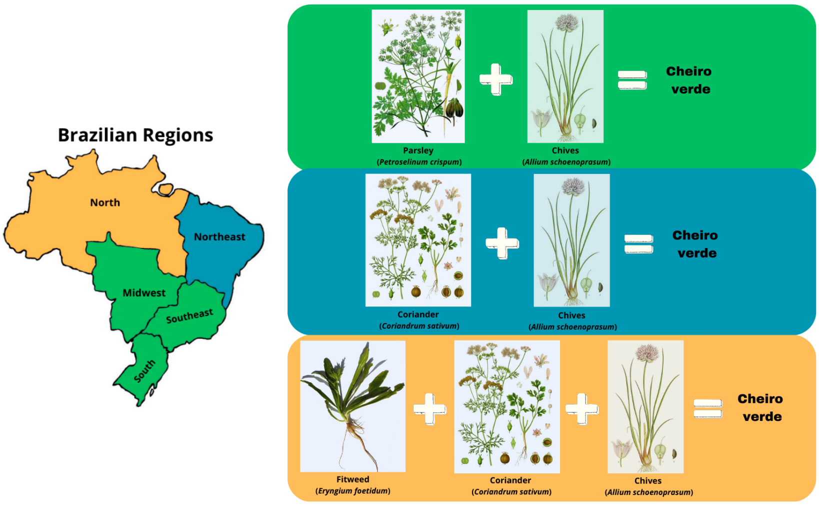


Figure 6: Herbs included in ‘cheiro verde” by Brazilian regions. Parsley image source: [71]; Coriander image source: [72]; Chives image source: [73]; Fitweed image source: [74]. All images reused in this figure are licensed under the public domain on Wikimedia Commons.

Listing 6: ‘Cheiro verde’ representation in APTO

```
Class: ‘Cheiro verde’
    SubClassOf: ‘Culinary herbs’
    ’Equivalent to’(Coriander and Chives) or
    (Coriander and Chives and Fitweed) or
    (Parsley and Chives)
...
Class: ‘Coriander’
    ‘member of taxon’ exactly 1 ‘Coriandrum sativum’

Class: ‘Chives’
    ‘member of taxon’ exactly 1  ‘Allium schoenoprasum’

Class: ‘Fitweed’
    ‘member of taxon’ exactly 1 ‘Eryngium foetidum’

Class: ‘Parsley’
    ‘member of taxon’ exactly 1 ‘Petroselinum crispum’
```

8. Final Considerations

This study explored two approaches for integrating taxonomic data into APTO, leveraging ChatGPT with the BrowserOP plugin (Approach 1) and a Python algorithm developed with ChatGPT's assistance (Approach 2). Each approach demonstrated unique strengths and weaknesses, highlighting the potential and challenges of integrating Artificial Intelligence and automation in ontology development.

Approach 1 utilized ChatGPT for direct interaction with the GBIF API, demonstrating its capability to handle complex queries, retrieve taxonomic information, and verify synonym relationships. However, scalability limitations became evident as dataset size increased, leading to longer processing times and issues such as hallucinated URIs and incomplete tasks. The subsequent discontinuation of GPT plugins further limited the reproducibility of this approach.

In contrast, Approach 2 addressed these limitations by employing a Python algorithm to process more extensive lists of species names efficiently. This approach successfully handled a list of 74 plant species in just over 2.5 minutes while maintaining consistent class-subclass relationships and avoiding redundancy in taxonomic hierarchies. However, it was sensitive to typographical errors, necessitating a pre-processing step with ChatGPT for name validation.

The study underscores the complementary strengths of generative AI and algorithmic methods in ontology development. ChatGPT proved effective for tasks such as synonym detection, taxonomic hierarchy generation, and OWL code creation. Its ability to learn from errors and refine subsequent outputs exemplifies its potential for interactive and adaptive ontology design. Meanwhile, the Python algorithm provided the scalability and robustness required for handling larger datasets, demonstrating the practicality of combining AI-driven tools with structured programming techniques.

Challenges remain, particularly regarding dependency on external tools like ChatGPT, which may evolve unpredictably, as seen with the discontinuation of GPT plugins. Additionally, addressing semantic variations, such as those exemplified by the concepts ‘Pimenta’ and ‘Cheiro Verde,’ requires contextual modeling that goes beyond simple automation.

The workflow developed in this study is adaptable beyond the agricultural domain and can be applied to fields such as biodiversity informatics, ecological modeling, and conservation planning. Future work should explore fully automated solutions for error detection and validation, potentially incorporating machine learning models to reduce reliance on manual intervention. A complementary algorithm should be developed to periodically verify all species names in the ontology and update them in alignment with the latest version of the GBIF Backbone Taxonomy, for each new ontology version release.

In conclusion, this paper demonstrates how combining AI-driven tools can streamline ontology engineering, address domain-specific challenges, and enhance scalability and efficiency.

Data and Code Availability

All data and codes associated with this paper have been made openly available on Zenodo:

1. Soares, F. M., Saraiva, A. M., Ferreira Pires, L., da Silva Santos, L. O. B., de Abreu Moreira, D., Corrêa, F. E., Braghetto, K. R., Pignatari Drucker, D., & Botazzo Delbem, A. C. (2025). Supporting Data for “Exploring ChatGPT-4 for Transforming Taxonomic Data into OWL: Lessons Learned and Implications for Ontology Development” [Data set]. Version 2. Zenodo. `https://doi.org/10.5281/zenodo.14982527`
2. Soares, F. M. (2025). GBIF Taxonomy OWLizer. [Software]. Version 2. Zenodo. `https://doi.org/10.5281/zenodo.14984136`

Author Contributions

F.M.S. conceived and conducted the experiments and drafted the original manuscript. L.F.P., L.O.B.S.S., and A.M.S. supervised the study. F.M.S., D.A.M., F.E.C., K.R.B., D.P.D., and A.C.B.D. contributed to the formal analysis. All authors reviewed and approved the final text.

Acknowledgements

F.M.S. acknowledges support from the Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) under grant numbers 21/15125-0 and 22/08385-8. K.R.B. received funding from FAPESP (Grant n. 2023/00779-0). F.M.S., D.P.D., K.R.B., F.E.C., D.A.M., A.C.B.D., and A.M.S. thank the Center for Artificial Intelligence (C4AI), a partnership of USP, IBM, and FAPESP (Grant 2019/07665-4).

Competing Interests

The authors declare that they have no competing interests.

Declaration of AI Use in the Writing Process

During the preparation of this work the authors used ChatGPT-4o with Scholar AI and Grammarly Education in order to review the text flow, grammar, spelling, and to eliminate redundant text. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

References

[1] S. Yu, X. Youming, and Z. Li, "Formal contexts in ontologies," in *2011 6th International Conference on Computer Science and Education (ICCSE)*, 2011, pp. 214–217.

[2] J. Sarika and M. Sanju, "Knowledge representation with ontology tools & methodology," *International Journal of Computer Applications*, pp. 1–5, 2014.

[3] V. M. P. Vidal, J. a. C. Pinheiro, E. R. Sacramento, J. A. Fernandes De Macêdo, and B. F. Lóscio, "An ontology-based framework for heterogeneous data sources integration," *Revista de Informática Teórica e Aplicada*, vol. 16, no. 2, pp. 61–64, Mar. 2010. [Online]. Available: https://seer.ufrgs.br/rita/article/view/rita_v16_n2_p61

[4] E. Femi Aminu, I. O. Oyefolahan, M. Bashir Abdullahi, and M. T. Salaudeen, "A review on ontology development methodologies for developing ontological knowledge representation systems for various domains," *International Journal of Information Engineering and Electronic Business*, vol. 12, no. 2, p. 28–39, Apr. 2020.

[5] M. L. D. A. Campos, M. L. M. Campos, A. M. R. Dávila, H. E. Gomes, L. M. Campos, and L. Lira, "Methodological aspects concerning the reuse of ontologies: a study based on genomic annotations in the domain of trypanosomatides," *RECIIS*, vol. 3, no. 1, p. 243/257, Mar. 2009. [Online]. Available: https://www.reciis.icict.fiocruz.br/index.php/reciis/article/view/820/1696

[6] M. E. Ruiz, O. Bodenreider, E. Little, and P. Srinivasan, "Ontological research and its applications to the biomedical domain," *Proceedings of the American Society for Information Science and Technology*, vol. 42, no. 1, p. meet.14504201198, Jan. 2006.

[7] C. Gillott, *Taxonomy and Systematics*. Dordrecht: Springer Netherlands, 1995, p. 91–112. [Online]. Available: http://link.springer.com/10.1007/978-94-017-4380-8_4

[8] L. Peruzzi, "Advances in plant taxonomy and systematics," *Biology*, vol. 12, no. 4, p. 570, Apr. 2023. [Online]. Available: https://www.mdpi.com/2079-7737/12/4/570

[9] M. Kuntner and I. Agnarsson, "Are the linnean and phylogenetic nomenclatural systems combinable? recommendations for biological nomenclature," *Systematic Biology*, vol. 55, no. 5, p. 774–784, Oct. 2006. [Online]. Available: https://academic.oup.com/sysbio/article/55/5/774/1664874

[10] G. Rouhan and M. Gaudeul, *Plant Taxonomy: A Historical Perspective, Current Challenges, and Perspectives*. New York, NY: Springer US, 2021, vol. 2222, p. 1–38. [Online]. Available: https://link.springer.com/10.1007/978-1-0716-0997-2_1

[11] X. Feng, X. Wu, and H. Meng, "Ontology-grounded automatic knowledge graph construction by llm under wikidata schema," in *Proceedings of the KDD Workshop on Human-Interpretable AI 2024 (HI-AI 2024), co-located with 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2024)*, ser. CEUR Workshop Proceedings, vol. 3841. CEUR-WS.org, August 26 2024, pp. 117–135. [Online]. Available: https://ceur-ws.org/Vol-3841/Paper19.pdf

[12] M. J. Saeedizade and E. Blomqvist, *Navigating Ontology Development with Large Language Models*. Cham: Springer Nature Switzerland, 2024, vol. 14664, p. 143–161. [Online]. Available: https://link.springer.com/10.1007/978-3-031-60626-7_8

[13] T. Aggarwal, A. Salatino, F. Osborne, and E. Motta, "Large language models for scholarly ontology generation: An extensive analysis in the engineering field," *arXiv*, 2024. [Online]. Available: https://arxiv.org/abs/2412.08258

[14] A. Zaitoun, T. Sagi, S. Wilk, and M. Peleg, "Can large language models augment a biomedical ontology with missing concepts and relations?" *arXiv*, 2023. [Online]. Available: https://arxiv.org/abs/2311.06858

[15] M. Trajanoska, R. Stojanov, and D. Trajanov, "Enhancing knowledge graph construction using large language models," *arXiv*, 2023. [Online]. Available: https://arxiv.org/abs/2305.04676

[16] J. C. dos Reis and A. G. Regino, "Generating e-commerce related knowledge graph from text: Open challenges and early results using llms," in *Joint Proceedings of the 3rd International Workshop on Knowledge Graph Generation from Text (TEXT2KG) and Data Quality meets Machine Learning and Knowledge Graphs (DQMLKG)*, ser. CEUR Workshop Proceedings, vol. 3747. CEUR-WS.org, May 26-30 2024, p. paper 11. [Online]. Available: https://ceur-ws.org/Vol-3747/text2kg_paper11.pdf

[17] R. Amini, S. S. Norouzi, P. Hitzler, and R. Amini, "Towards complex ontology alignment using large language models," *arXiv*, 2024. [Online]. Available: https://arxiv.org/abs/2404.10329

[18] N. Matentzoglu, J. H. Caufield, H. B. Hegde, J. T. Reese, S. Moxon, H. Kim, N. L. Harris, M. A. Haendel, and C. J. Mungall, "Mappergpt: Large language models for linking and mapping entities," *arXiv*, 2023. [Online]. Available: https://arxiv.org/abs/2310.03666

[19] G. Wu, C. Ling, I. Graetz, and L. Zhao, "Ontology extension by online clustering with large language model agents," *Frontiers in Big Data*, vol. 7, p. 1463543, Oct. 2024. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fdata.2024.1463543/full

[20] L. van Hees, "Exploring the utility of large language models for achieving semantic interoperability in data ecosystems," Master Thesis, Utrecht University, 2024. [Online]. Available: https://studenttheses.uu.nl/handle/20.500.12932/48039

[21] Y. Zhang, M. Zhong, S. Ouyang, Y. Jiao, S. Zhou, L. Ding, and J. Han, "Automated mining of structured knowledge from text in the era of large language models," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ser. KDD '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 6644–6654. [Online]. Available: https://doi.org/10.1145/3637528.3671469

[22] M. Hofer, D. Obraczka, A. Saeedi, H. Köpcke, and E. Rahm, "Construction of knowledge graphs: Current state and challenges," *Information*, vol. 15, no. 8, p. 509, Aug. 2024. [Online]. Available: https://www.mdpi.com/2078-2489/15/8/509

[23] J. Chen, H. Dong, J. Hastings, E. Jiménez-Ruiz, V. López, P. Monnin, C. Pesquita, P. Škoda, and V. Tamma, "Knowledge Graphs for the Life Sciences: Recent Developments, Challenges and Opportunities," *Transactions on Graph Data and Knowledge*, vol. 1, no. 1, pp. 5:1–5:33, 2023. [Online]. Available: https://drops.dagstuhl.de/entities/document/10.4230/TGDK.1.1.5

[24] S. Tsaneva, S. Vasic, and M. Sabou, "Llm-driven ontology evaluation: Verifying ontology restrictions with chatgpt," in *Data Quality meets Machine Learning and Knowledge Graphs: DQMLKG Workshop at ESWC 2024*, 2024, in press.

[25] O. Perera and J. Liu, "Exploring large language models for ontology learning," *Issues in Information Systems*, vol. 25, no. 4, pp. 299–310, 2024. [Online]. Available: https://doi.org/10.48009/4_iis_2024_124

[26] O. B. Mulayim, L. Paul, M. Pritoni, A. K. Prakash, M. Sudarshan, and G. Fierro, "Large language models for the creation and use of semantic ontologies in buildings: Requirements and challenges," in *Proceedings of the 11th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, ser. BuildSys '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 312–317. [Online]. Available: https://doi.org/10.1145/3671127.3698792

[27] F. Neuhaus, "Ontologies in the era of large language models – a perspective," *Applied Ontology*, vol. 18, no. 4, p. 399–407, Dec. 2023. [Online]. Available: https://journals.sagepub.com/doi/full/10.3233/AO-230072

[28] I. Subirats-Coll, K. Kolshus, A. Turbati, A. Stellato, E. Mietzsch, D. Martini, and M. Zeng, "Agrovoc: The linked data concept hub for food and agriculture," *Computers and Electronics in Agriculture*, vol. 196, p. 105965, 2022.

[29] M. Ambrosio Telles, C. Jonquet, B. Teixeira de Almeida, J. Daltio, C. Maki Takemura, L. Henrique Mendonça de Oliveira, and M. d. C. F. Alencar, "Embrapa's contributions to integrate brazilian agricultural vocabularies: Agrotermos in AgroPortal," in *CEUR Workshop Proceedings*, vol. 3905, no. 8, C. M. Fonseca and V. Zamborlini and E. Cardoso and J. L. Emygdio. Vitória, Brazil: CEUR Proceedings, Oct. 2024. [Online]. Available: https://hal.science/hal-04928168

[30] C. Caracciolo, A. Stellato, A. Morshed, G. Johannsen, S. Rajbhandari, Y. Jaques, and J. Keizer, "The agrovoc linked dataset," *Semantic Web*, vol. 4, no. 3, p. 341–348, 2013.

[31] I. Pierozzi Júnior, M. C. Visoli, M. I. F. Souza, L. M. S. Cunha, I. Vacari, and T. Z. Torres, *Engenharia da informação: contribuições para a agricultura digital*. Brasília, DF: Embrapa, 2020, ch. 8, pp. 192–217. [Online].

Available: http://www.alice.cnptia.embrapa.br/handle/doc/1126265
[32] Food and A. O. of the United Nations, *AGROVOC 4: AGROVOC editorial guidelines*. FAO, Feb. 2023. [Online]. Available: http://www.fao.org/documents/card/en/c/cc0647en
[33] G. Secretariat, "Prochilodus cearensis steindachner, 1911 in gbif backbone taxonomy," https://doi.org/10.15468/39omei, 2023, accessed via GBIF.org on 2024-06-19.
[34] R. d. A. Falbo, "Sabio: Systematic approach for building ontologies," *Onto. Com/odise@ Fois*, vol. 1301, 2014.
[35] M. Krötzsch, "Ontologies for knowledge graphs?" in *Proceedings of the 30th International Workshop on Description Logics (DL 2017)*, ser. CEUR Workshop Proceedings, A. Artale, B. Glimm, and R. Kontchakov, Eds., vol. 1879. Montpellier, France: CEUR-WS.org, July 18–21 2017. [Online]. Available: https://ceur-ws.org/Vol-879/invited2.pdf
[36] M. Krötzsch and V. Thost, *Ontologies for Knowledge Graphs: Breaking the Rules*. Cham: Springer International Publishing, 2016, vol. 9981, p. 376–392. [Online]. Available: https://link.springer.com/10.1007/978-3-319-46523-4_23
[37] N. Mihindukulasooriya, S. Tiwari, C. F. Enguix, and K. Lata, *Text2KGBench: A Benchmark for Ontology-Driven Knowledge Graph Generation from Text*. Cham: Springer Nature Switzerland, 2023, vol. 14266, p. 247–265. [Online]. Available: https://link.springer.com/10.1007/978-3-031-47243-5_14
[38] A. Hogan, E. Blomqvist, M. Cochez, C. D'amato, G. D. Melo, C. Gutierrez, S. Kirrane, J. E. L. Gayo, R. Navigli, S. Neumaier, A.-C. N. Ngomo, A. Polleres, S. M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, and A. Zimmermann, "Knowledge graphs," *ACM Comput. Surv.*, vol. 54, no. 4, Jul. 2021. [Online]. Available: https://doi.org/10.1145/3447772
[39] S. Pan, L. Luo, Y. Wang, C. Chen, J. Wang, and X. Wu, "Unifying large language models and knowledge graphs: A roadmap," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–20, 2024.
[40] A. Kumar, A. Pandey, R. Gadia, and M. Mishra, "Building knowledge graph using pre-trained language model for learning entity-aware relationships," in *2020 IEEE International Conference on Computing, Power and Communication Technologies (GUCON)*, 2020, pp. 310–315.
[41] A. Bosselut, H. Rashkin, M. Sap, C. Malaviya, A. Celikyilmaz, and Y. Choi, "Comet: Commonsense transformers for automatic knowledge graph construction," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, 2019, p. 4762–4779. [Online]. Available: https://www.aclweb.org/anthology/P19-1470
[42] Center for Advanced Studies on Applied Economics (CEPEA), "Cepea agricultural prices," https://cepea.esalq.usp.br/en, 2024, accessed: 2024-06-13.
[43] Instituto de Pesquisa Econômica Aplicada (IPEA), "Ipeadata: indicadores macroeconomicos," http://www.ipeadata.gov.br/Default.aspx, 2024, accessed: 2024-06-13.
[44] Companhia Nacional de Abastecimento (CONAB), "Conab - preços agropecuários," https://www.conab.gov.br/info-agro/precos, 2024, accessed: 2024-06-13.
[45] Global Biodiversity Information Facility, "Gbif home page," https://www.gbif.org/, 2024, accessed: 28-04-2024.
[46] B. Sterner, S. Elliott, E. E. Gilbert, and N. M. Franz, "Unified and pluralistic ideals for data sharing and reuse in biodiversity," *Database*, vol. 2023, p. baad048, Jul. 2023. [Online]. Available: https://academic.oup.com/database/article/doi/10.1093/database/baad048/7226274
[47] Convention on Biological Diversity and Global Biodiversity Information Facility, "Big data for biodiversity: Global biodiversity information facility surpasses one billion records," https://www.cbd.int/doc/press/2018/pr-2018-07-09-cbd-gbif-en.pdf, 2018, accessed: 28-04-2024.
[48] GBIF Secretariat, "Gbif backbone taxonomy," https://doi.org/10.15468/39omei, 2023, accessed via GBIF.org on 2024-03-04.
[49] ——, "Gbif api reference," https://techdocs.gbif.org/en/openapi/, accessed via GBIF.org on 2024-03-04.
[50] D. Hobern, S. K. Barik, L. Christidis, S. T.Garnett, P. Kirk, T. M. Orrell, T. Pape, R. L. Pyle, K. R. Thiele, F. E. Zachos, and O. Bánki, "Towards a global list of accepted species vi: The catalogue of life checklist," *Organisms Diversity and Evolution*, vol. 21, no. 4, p. 677–690, Dec. 2021. [Online]. Available: https://link.springer.com/10.1007/s13127-021-00516-w
[51] W. Greuter, D. L. Hawksworth, J. McNeill, M. A. Mayo, A. Minelli, P. H. A. Sneath, B. J. Tindall, P. Trehane, and P. Tubbs, "Draft biocode (1997): The prospective international rules for the scientific names of organisms," *Taxon*, vol. 47, no. 1, pp. 127–150, 1998.
[52] J. E. Winston, "Twenty-First Century Biological Nomenclature—The Enduring Power of Names," *Integrative and Comparative Biology*, vol. 58, no. 6, pp. 1122–1131, 09 2018.
[53] S. Okasha, "Species and classification," in *Philosophy of Biology: A Very Short Introduction*. Oxford University Press, 11 2019.
[54] L. Giray, "Prompt engineering with chatgpt: A guide for academic writers," *Annals of Biomedical Engineering*, vol. 51, no. 12, p. 2629–2633, Dec. 2023.
[55] OpenAI, "Meet browserop: our chatgpt plugin that lets you search up-to-date information smoothly

and efficiently," https://community.openai.com/t/meet-browserop-our-chatgpt-plugin-that-lets-you-search-up-to-date-information-smoothly-and-efficiently/304018, 2023, accessed: 01-05-2024.

[56] W. Ride, H. Cogger, C. Dupuis, O. Kraus, A. Minelli, F. Thompson, and P. Tubbs, Eds., *International Code of Zoological Nomenclature*, 4th ed. London: The International Trust for Zoological Nomenclature, 2012. [Online]. Available: https://www.iczn.org/the-code/the-code-online/

[57] N. J. Turland, J. H. Wiersema, F. R. Barrie, W. Greuter, D. L. Hawksworth, P. S. Herendeen, S. Knapp, W.-H. Kusber, D.-Z. Li, K. Marhold, T. W. May, J. McNeill, A. M. Monro, J. Prado, M. J. Price, and G. F. Smith, Eds., *International Code of Nomenclature for algae, fungi, and plants (Shenzhen Code)*, ser. Regnum Vegetabile. Glashütten: Koeltz Botanical Books, 2018, vol. 159, adopted by the Nineteenth International Botanical Congress Shenzhen, China, July 2017.

[58] Z. Wang, G. Zhang, K. Yang, N. Shi, W. Zhou, S. Hao, G. Xiong, Y. Li, M. Y. Sim, X. Chen, Q. Zhu, Z. Yang, A. Nik, Q. Liu, C. Lin, S. Wang, R. Liu, W. Chen, K. Xu, D. Liu, Y. Guo, and J. Fu, "Interactive natural language processing," 2023. [Online]. Available: https://arxiv.org/abs/2305.13246

[59] T. P. Nagarhalli, S. Mhatre, S. Patil, and P. Patil, "The review of natural language processing applications with emphasis on machine learning implementations," in *2022 International Conference on Electronics and Renewable Systems (ICEARS)*. Tuticorin, India: IEEE, Mar. 2022, p. 1353–1358. [Online]. Available: https://ieeexplore.ieee.org/document/9752326/

[60] T. Shaik, X. Tao, Y. Li, C. Dann, J. McDonald, P. Redmond, and L. Galligan, "A review of the trends and challenges in adopting natural language processing methods for education feedback analysis," *IEEE Access*, vol. 10, p. 56720–56739, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9781308/

[61] O. Bulut and T. Wongvorachan, "Feedback generation through artificial intelligence," *The Open/Technology in Education, Society, and Scholarship Association Conference*, vol. 2, no. 1, p. 1–9, Dec. 2022. [Online]. Available: https://conference.otessa.org/index.php/conference/article/view/125

[62] Royal Botanic Gardens, Kew, "Plants of the World Online," http://www.plantsoftheworldonline.org/, 2024, retrieved 14 March 2024.

[63] H. W. Rickett and W. H. Camp, "The nomenclature of hybrids," *Bulletin of the Torrey Botanical Club*, vol. 75, no. 5, p. 496–501, 1948.

[64] N. J. Turland, J. H. Wiersema, F. R. Barrie, W. Greuter, D. L. Hawksworth, P. S. Herendeen, S. Knapp, W.-H. Kusber, D.-Z. Li, K. Marhold, T. W. May, J. McNeill, A. M. Monro, J. Prado, M. J. Price, and G. F. Smith, "Chapter H: Names of Hybrids," in *International Code of Nomenclature for algae, fungi, and plants (Shenzhen Code)*. Koeltz Botanical Books, 11 2019. [Online]. Available: https://www.iapt-taxon.org/nomen/pages/main/art_h3.html

[65] Food and Agriculture Organization of the United Nations, "Pepper," in *Agrovoc*. Food and Agriculture Organization of the United Nations, 2011, created 11/20/11, last modified 8/28/23. [Online]. Available: http://aims.fao.org/aos/agrovoc/c_25495

[66] Embrapa, "Pimenta," in *Agrotermos: Espaço conceitual da Embrapa*. Embrapa, 2017. [Online]. Available: https://sistemas.sede.embrapa.br/agrotermos/resources/bcf87c03ae11d3d7a749a75c87b4649a

[67] Food and A. O. of the United Nations, "Chillies," in *Agrovoc*. FAO, 2012, accessed: 2024-08-18. [Online]. Available: http://aims.fao.org/aos/agrovoc/c_9795

[68] Embrapa, "Condimento picante," in *Agrotermos: Espaço conceitual da Embrapa*. Embrapa, 2017, accessed: 2024-08-18. [Online]. Available: https://sistemas.sede.embrapa.br/agrotermos/resources/22538a47e4cb9e2ca6a5ed5d3fc59ce3

[69] Brasil. Ministério da Saúde. Secretaria de Políticas de Saúde. Coordenação-Geral da Política de Alimentação e Nutrição, *Alimentos regionais brasileiros*, ser. Série F. Comunicação e Educação em Saúde. Brasília: Ministério da Saúde, 2002, no. 21.

[70] G. Menezes, "Cheiro-verde é coentro? entenda a combinação de temperos," September 2021. [Online]. Available: https://www.uol.com.br/nossa/noticias/redacao/2021/09/17/cheiro-verde-e-coentro-entenda-o-que-e-o-tempero.htm?cmpid=copiaecola

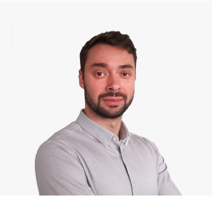
[71] F. E. Köhler, *Petroselinum crispum*. WikiMedia Commons, 1897. [Online]. Available: https://commons.wikimedia.org/wiki/File:Petroselinum_crispum_-_K%C3%B6hler%E2%80%93s_Medizinal-Pflanzen-103.jpg

[72] ——, *Coriandrum sativum*. WikiMedia Commons, 1897. [Online]. Available: https://commons.wikimedia.org/wiki/File:Coriandrum_sativum_-_K%C3%B6hler%E2%80%93s_Medizinal-Pflanzen-193.jpg

[73] C. A. M. Lindman, *Allium schoenoprasum L.* Stockholm: WikiMedia Commons, 1917–1926. [Online]. Available: https://commons.wikimedia.org/wiki/File:386_Allium_schoenoprasum.jpg

[74] DXLINH, "Cayngotau," 2022. [Online]. Available: https://commons.wikimedia.org/wiki/File:Cayngotau.jpg

Author Biography



Filipi Miranda Soares, M.Sc. (Knowledge Management), is a Ph.D. candidate in Computer Engineering at the University of São Paulo, Brazil, and in Computer Science at the University of Twente, the Netherlands, as part of a double degree program. His research focuses on integrating ontologies, Knowledge Graphs, and Generative Artificial Intelligence for enhanced agricultural data analysis.

ORCID: 0000-0002-0674-7960

Antonio Mauro Saraiva, Ph.D. (Electrical Engineering), is a Senior Full Professor at the Institute of Advanced Studies (IEA) at USP. From 1989 to 2024, he was a professor at the Department of Computer Engineering and Digital Systems (Polytechnic School - USP), serving as Full Professor from 2008 to 2024, until his retirement. He was Head of the Department (2014–2016), Associate Provost for Research at USP (2014–2016), and Advisor to the Research Provost (2016–2020). He chaired the Research Committees at Poli-USP (2011–2016) and IEA-USP (2018–2020).

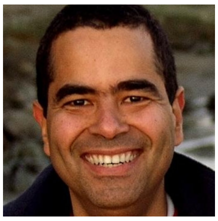
ORCID: 0000-0003-2283-1123

Luís Ferreira Pires, Ph.D. He obtained his B.Sc. degree in Electronics from the ’Instituto Tecnológico de Aeronáutica’ (São José dos Campos, Brazil) in 1983 and, his M.Sc. degree in Electrical Engineering from the ’Escola Politécnica da Universidade de São Paulo’ (São Paulo, Brazil), in the Computer Networks specialisation, in 1989. He obtained his PhD degree from the Faculty of Electrical Engineering of the University of Twente, Enschede, the Netherlands, in September 1994. Since 1988 he is working at the University of Twente (Enschede, The Netherlands), initially as research fellow of the Faculty of Computer Science, from 1992 to 1994 as Assistant Professor of the Faculty of Electrical Engineering and since 1994 as Associate Professor of the Faculty of Electrical Engineering, Mathematics and Computer Science. He is a member of the Services & Cyber-Security (SCS) group.

ORCID: 0000-0001-7432-7653

Luiz Olavo Bonino da Silva Santos, Ph.D. He is an Associate Professor at the Semantic, Cybersecurity and Service group at the University of Twente and at the BioSemantics group, in the department of Human Genetics at the Leiden University Medical Center. His main research topics are the FAIR principles, semantic interoperability, ontology-driven conceptual modelling and intelligent systems design.

ORCID: 0000-0002-1164-1351



Dilvan de Abreu Moreira, Ph.D. (Electronics Engineering). PostDoc in Biomedical Informatics from Stanford University (2008 - USA), Ph.D. in Electronics Engineering from University of Kent at Canterbury (1995 - UK), M.Sc. in Microelectronics from State University of Campinas (1991 - Brazil), B.S. on Electric Engineering from Federal University of Bahia - UFBa (1988 - Brazil). Currently Associate Professor (tenured) at University of São Paulo - USP (since 1997). Have more than 20 years experience in software research and development: Web based distributed and client/server applications, including technologies such as Web services, ontologies (Semantic Web OWL), C, C++, Clojure, Groovy and Java languages on Linux, Mac and Windows. Held positions managing research laboratory operations.

ORCID: 0000-0002-4801-2225

Fernando Elias Corrêa, Ph.D. Phd. is a Data Science specialist and holds both master's and PhD degrees in Computer Engineering from São Paulo University (USP). Currently, is a Postdoctoral Researcher in data management applied to agribusiness at the Center for Artificial Intelligence – C4AI. His research is focused on artificial intelligence, Spatial Temporal and Data management.

ORCID: 0000-0002-2744-6104

Kelly Rosa Braghetto, Ph.D., is an Assistant Professor in the Department of Computer Science at the Institute of Mathematics and Statistics, University of São Paulo. She earned her Ph.D. (2011) and M.Sc. (2006) in Computer Science from the University of São Paulo. Her primary research interests include data engineering, data integration, scientific data management, and smart cities.

ORCID: 0000-0001-6218-6849

Debora Pignatari Drucker, Ph.D. (Environment and Society) is an analyst at Embrapa Digital Agriculture, a Brazilian Agricultural Research Corporation Research Center. She coordinates the GO FAIR Brazil Agriculture Thematic Network and is one of the co-chairs of the Research Data Alliance IGAD Community of Practice (Improving Global Agricultural Data). She is also an expert at the Data and Knowledge Task Force of the Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem Services (IPBES). Her major interest is in mobilizing high-quality data to support scientific research and decision-making.

ORCID: 0000-0003-4177-1322

Alexandre Cláudio Botazzo Delbem, Ph.D. Full Professor at the Department of Computer Systems of the Institute of Mathematical and Computer Sciences in the University of São Paulo (ICMC-USP). Delbem investigates computer-based solutions to work with real-world problems that can be modeled as cyber-physical systems. Investigations focus on multidisciplinary applications in fields such as: power restoration after blackouts in large-scale electrical networks, population dynamics and integrated projects of supply and attending networks involving agribusiness, healthcare, environment and social assistence.

ORCID: 0000-0003-1810-1742