DYNAMO: Dynamic Ontology Extension for Augmenting Chatbot Intelligence through BabelNet

Amalia Georgoudi
Information Technologies Institute
Centre for Research and Technology Hellas
Thessaloniki, Greece
ageorgoudi@iti.gr

Georgios Meditskos
School of Informatics
Aristotle University of Thessaloniki
Thessaloniki, Greece
gmeditsk@csd.auth.gr

Thanassis Mavropoulos
Information Technologies Institute
Centre for Research and Technology Hellas
Thessaloniki, Greece
mavrathan@iti.gr

Stefanos Vrochidis
Information Technologies Institute
Centre for Research and Technology Hellas
Thessaloniki, Greece
stefanos@iti.gr

Ioannis Kompatsiaris
Information Technologies Institute
Centre for Research and Technology Hellas
Thessaloniki, Greece
ikom@iti.gr

***Abstract*—Dynamic ontology extension, a real-time ontology extension process, facilitates continuous learning and adaptation to new knowledge and evolving domains. The ability to dynamically add concepts, relationships, and axioms allows intelligent agents and knowledge-based systems to stay up-to-date and responsive. This paper presents a novel approach to dynamically extend the ontology of a chatbot's Knowledge Base by leveraging BabelNet, a multilingual encyclopedic dictionary and semantic network. Using BabelNet's semantic relations, such as hyperonyms, hyponyms, and holonyms, we focus on enriching the ontology with user profile information, enabling knowledge inference and personalized interactions. Through repeated interactions, the chatbot increases its level of intelligence, inferring new knowledge and asking targeted questions to users, resulting in effective interactions and increased user satisfaction.**

***Index Terms*—Dynamic Ontology Extension; Ontology; Chatbot.**

I. INTRODUCTION

Ontologies play a vital role in improving the efficiency and effectiveness of retrieving information and representing knowledge online. Ontologies serve as organized structures for capturing knowledge and establishing a shared language to describe concepts, relationships, and properties within specific domains. Through explicit definitions of terms and their relationships, ontologies enable seamless integration, searching, and reasoning of information. They facilitate the creation of machine-readable and machine-understandable representations, fostering interoperability and data exchange across diverse systems and domains.

A chatbot is an Artificial Intelligence (AI) program specifically created to mimic human conversation and engage with users through interfaces that use text or voice. By incorporating ontologies into a Chatbot's Knowledge Base (KB), the chatbot would have access to a rich semantic network of concepts, relationships, and properties within a specific domain. Ontologies define the vocabulary and the meaning of terms, enabling the chatbot to understand user input and generate relevant and accurate responses.

Dynamic ontology extension is the process of extending or updating an existing ontology in real-time, depending on new information or shifting requirements. In order to enable the ontology to change and adapt to new knowledge or evolving domains, it entails dynamically adding new concepts, relationships, attributes, or axioms to it. Intelligent agents, semantic web applications, and knowledge-based systems all heavily depend on dynamic ontology extension in order to continuously learn, adapt, and incorporate new information.

Through the dynamic extension of the chatbot's KB ontology, based on user inputs, the chatbot gradually enhances its intelligence with each interaction. With the ability to expand its ontology in real-time, the chatbot becomes increasingly proficient at understanding user needs and providing tailored responses, leading to a more natural and effective user experience.

In this paper, we present our method for the dynamic ontology extension of a chatbot's KB ontology by utilizing BabelNet [11]. BabelNet is a public multilingual encyclopedic dictionary and a semantic network, in which every entity is connected with other entities through semantic relations, such as hypernyms, hyponyms and holonyms. We utilized those relations and we created a pattern, in order to automatically extend the ontology with more information regarding the users profiles, enabling inferencing knowledge, enhancing their understanding, and drawing conclusions. The goal is to enrich the situational awareness of the chatbot every time it interacts with the user, reducing the time required for the interaction and increasing user satisfaction. The proposed method extends the schema of the ontology with new classes and properties and also populates the ontology by adding new instances and forming semantic relationships and links between them.

The remainder of the paper is organized as follows. Section II presents the related work. In Section III the proposed method is introduced, while in Section IV a simulation example is presented. In Section V the evaluation procedure is presented. Section VI concludes and gives directions for future work.

II. Background and Related Work

The following subsections provide a brief overview of Knowledge Representation frameworks, Ontology-based Chatbots and existing Dynamic Ontology Extension approaches.

A. Knowledge Representation

An ontology is a structured framework for organizing information that provides a formal and explicit specification of a commonly recognized formulation of a field of interest. The representation of knowledge as a set of concepts, relationships, and properties is part of this formatting. A straightforward knowledge representation language known as the Resource Description Framework (RDF) [1] intends to standardize metadata and usage descriptions for Web-based resources. A group of subject-predicate-object triples serve as the fundamental building unit of RDF. To display richer and more complicated information about objects, groups of things, and relationships between them, the Web Ontology Language (OWL) [2], a collection of knowledge representation languages, was developed. OWL now extends previous Web standards for describing knowledge and is the official World Wide Web Consortium (W3C) recommendation for creating and sharing ontologies. An OWL ontology consists of: classes, properties and restrictions. Classes are the main element of an ontology and are used to describe a field's concepts. Properties are used to describe feature attributes, while restrictions are determining properties' confinement. Furthermore, an ontology has instances of classes and relationships between those.

B. Ontology-based Chatbots

Utilizing ontologies is a potential strategy that enables chatbots and conversational agents to comprehend and produce contextually relevant responses. In this section, many studies that present ontology-based chatbots and conversational agents are examined, highlighting their contributions and outlining their ramifications.

The SynchroBot [3] is a dialog system that can be connected to reliable and adaptable KBs and KGs for information extraction and that can use NLP tools to analyze user questions and NLG techniques to deliver appropriate answers. A previous ontology-based chatbot called OntBot used Hallili's method as its foundation. OntBot [4] uses a suitable mapping approach to convert ontologies and other knowledge into relational databases with a set of mapping rules.

The recent bibliography includes numerous proposed methods for introducing KGs to QA and AI chatbots in different domains, such as E-commerce [5], museums [6], healthcare systems [7]. A working model of Ontology based chatbot that handles queries from users for an E-commerce website, is proposed in [5]. This chatbot helps the user by mapping relationships of the various entities required by the user, thus providing detailed and accurate information there by overcoming the drawbacks of traditional chatbots. The author in [6] summarizes recent research on Knowledge Graph (KG)-based AI chatbot design and development for museums. The suggested MuBot approach gave museums the chance to develop chatbots for their visitors. The use of KGs for chatbot implementation raises issues with translating natural language dialogues

In [7], the suggested conversation agent, is built on an ontology-based knowledge model that enables flexible reasoning-driven dialogue planning as opposed to the use of predetermined dialogue scripts. Another comparable strategy was used to the healthcare industry with the intention of creating a framework that may help patients by giving them access to an AI chatbot with good conversational abilities and a substantial KB [9] [8]. MediBot [10] is another ontology-based chatbot created to facilitate access to information on drugs and their risks easily and directly to Portuguese speakers.

C. Dynamic Ontology Extension Frameworks

Ontology evolution is a subfield of ontology change which refers to the problem of transforming an ontology in response to a certain need [12]. Specifically, ontology evolution consists of transforming an ontology or incorporating new information in an existing ontology, in a way that it satisfies the users and describes the knowledge domain, while maintaining its consistency.

In survey [13], the authors outline the process of ontology learning and categorizes ontology learning techniques into three classes: linguistics [14], statistical [19] [16], and logical [19] that are based on reasoning rules. It examines the advantages and disadvantages of ontology evaluation techniques and it explores the applications and significance of ontology learning in different industries.

Linguistic-based approaches for ontology extension utilize NLP tasks, such as Named Entity Recognition, Part-Of-Speech Tagging and Dependency Parsing and they have been proposed for many different languages and domains. For instance, there are approaches for Spanish legal texts [15], French [17] and Chinese [18] documents.

A statistical and logical system, known as CRCTOL is proposed in [19]. CRCTOL is a system designed to automate the extraction of ontologies from domain-specific documents. The proposed system, CRCTOL, utilizes a comprehensive text parsing technique and incorporates a combination of statistical and lexico-syntactic methods and includes a a statistical, a word sense disambiguation and a rule-based algorithm.

In addition, some proposed methods utilize Deep Learning models, such as [20] [22]. In [20], a novel approach to automatically expand ontologies, specifically focusing on the ChEBI ontology, a well-established reference in the field of chemistry within life sciences, is presented. To achieve this, the authors utilized a deep learning model called ChemBERTa [21], which is built upon the Transformer architecture. The model was trained using the leaf node structures found in the ChEBI ontology along with their corresponding classes.

 ISBN: 978-1-68558-108-4

In [22], a bi-LSTM-based word extraction model based on character embedding is presented to extract the terms from a phrase for automatic ontology population.

Our approach distinguishes itself by leveraging publicly available resources, such as BabelNet and Wikidata. These resources play a vital role in shaping the classes and their hierarchical relationships within the ontology, employing semantic relations like hyponyms and holonyms. Furthermore, our approach utilizes dependency parsing, a process to grammatically analyze sentences, to establish direct and indirect data and object properties. Our methodology ensures that multilingual information is preserved, thereby enabling multilingual interactions. By incorporating these distinct elements, our approach presents a novel and robust framework for ontology development and facilitates enhanced linguistic and semantic interoperability.

III. Dynamic Ontology Extension Approach

This section provides a high level overview of the architectural design of the dynamic ontology extension approach. Our approach extends the schema of the ontology with new classes and properties and also populates the ontology by adding new instances and forming semantic relationships and links between them.

The method is built upon a language analysis task that specifically targets named entity recognition. The solution has been implemented utilizing the Python programming language, harnessing a range of libraries, including Owlready2, Babelnet, SPARQLWrapper, among others. Through this solution, the ontology is augmented by incorporating entities identified as persons, organizations, or geopolitical locations. This enriched information plays a pivotal role in enabling a chatbot to draw accurate conclusions and gain a deeper understanding of the users' background, thereby facilitating more effective interactions.

The approach encompasses four distinct phases, each serving a specific purpose in the overall ontology extension process. These phases are: the Disambiguation, the Formulation of Classes' Hierarchy, the Population of the Ontology and finally the Generation of Data and Object Properties.

A. Disambiguation

When an entity, mentioned in the speaker's utterances, is classified as a Named Entity, and contains a Babelnet id, the service will be enabled. The entity tag will constitute the classes, while the entities will constitute the instances. Three different cases are supported:

1) The class does not exist in the ontology.
2) The class already exists, but the specific instance does not exit.
3) Both the class and the specific instance already exist in the ontology.

The first case is when the class does not exist in the ontology. Our service creates the class, the necessary properties, and adds the new instance. In case that the class already exists, our service creates only the new instance of this class. Otherwise, if both the class and the specific instance exist, our service will not extend further by adding duplicates into the ontology.

Moreover, an additional disambiguation phase is conducted for entities categorized as geopolitical locations or organizations, utilizing BabelNet relations. Within the disambiguation phase for geopolitical location entities, the primary objective is to ascertain whether the entity represents a country or a city. In the case of a city, a further check is performed to determine if it serves as a capital. Additionally, for every entity that is tagged as an organization, the goal is to understand the type of the organization. For instance an organization may be a university, a hospital, a bank, etc. This disambiguation process relies on BabelNet synsets and ISA relationships to accomplish the aforementioned tasks.

B. Formulation of Classes' Hierarchy

During the second phase, the algorithm proceeds with the creation of classes and subclasses, following a hierarchy derived from BabelNet relationships. This process leverages the information acquired during the disambiguation phase, wherein hypernyms, hyponyms, and holonyms associated with each entity tag are utilized to effectively structure the classes within the ontology.

A hypernym relation refers to a hierarchical relationship where one entity is more general or broader in meaning than another entity. It signifies that the first entity encompasses or includes the second entity within its scope. Holonyms in BabelNet denote the connection between a complete entity and its component parts. A holonym represents the entirety of the entity, whereas its parts are identified as meronyms.

By utilizing these semantic relationships, the algorithm ensures a coherent and organized representation of the entities within the ontology, facilitating better categorization and classification. Figure 1 illustrates an example of the classes that were created form the Geopolitical Entities (GPE) instances "Greece" and "Sweden", by iteratively exploiting BabelNet relationships.

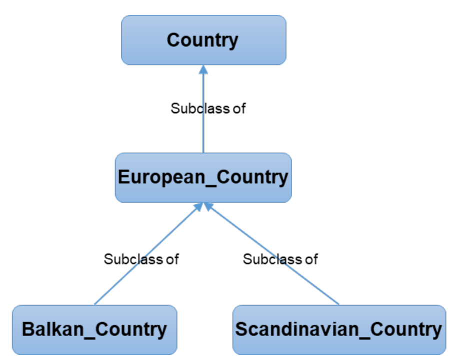

Fig. 1. Hierarchy of classes example.

C. Population of the Ontology

Following the successful construction of the ontology's classes, the third phase involves the population of the ontology. Within this phase, the entities will undergo classification as instances of specific classes, determined by their assigned named entity tags. Subsequently, the formation of links and connections between distinct entities will take place.

To avoid duplicate elements within the ontology, the service incorporates a mechanism that checks for the existence of classes and properties before creating new ones. For instance, if an entity is associated with the ”person” tag, a new class named ”person” will only be generated if it does not already exist in the ontology. Similarly, each time an entity is tagged as ”person,” a new instance will be added to the corresponding class, ensuring that instances are not duplicated. This approach maintains the integrity and coherence of the ontology by preventing redundant class and instance creation.

D. Generation of Data and Object Properties

For the data properties, we retrieve important information from the BabelNet in order to form the triples. First of all, each entity is associated with a data property that stores its unique BabelNet ID. Furthermore, we store the information in multiple languages, allowing the chatbot to engage in multilingual conversations effectively. Additionally, we enhance the data by incorporating supplementary details from Wikidata, utilizing SPARQL queries specifically tailored for each geopolitical location. This comprehensive approach ensures that the ontology encompasses diverse and enriched data, enabling the chatbot to provide accurate and contextually relevant information during interactions. Table I shows some examples of data properties.

TABLE I
DATA PROPERTIES

Data Property	*Description*	*Range*
official_languages	The official languages of a country are the languages that are recognized and used by the government for official business and communication	string
babelnet_id	The unique babelnet if of every entity	string
has_value	Gives the name in different languages	string
geo_lat	Geographical coordinates: Latitude	float
geo_long	Geographical coordinates: Longitude	float

Relations between entities in the ontology are also created. Also, inverse properties and transitive properties are created between two entities. Table II represents some object properties that can be created between different entities, such as cities and countries.

TABLE II
OBJECT PROPERTIES

Object Property	*Description*	*Domain*	*Range*
is_city_of	A city is in a country	City	Country
has_city	A country has a city	Country	City
is_in_city	An organization is in a city	Organization	City
Has_organization	A city has an organization	City	Organization

Furthermore, the ontology is extended with direct and indirect relations between the speaker and the entities, depending on what the user said. Figure 2 represents the created direct and indirect relationships between the speaker and the entities, based on the user’s utterance *”I live in Thessaloniki with my friend Maria.”*. In order to achieve this, we parse relations in the dependency tree to connect indirect entities.

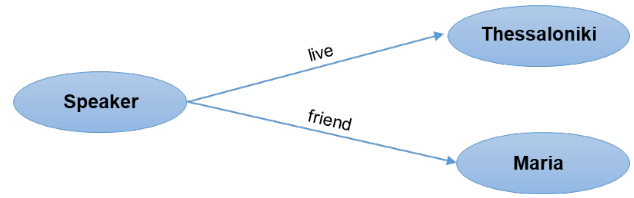

Fig. 2. Direct and indirect relations with the speaker.

IV. EXAMPLE USE CASES

Below, we will describe two simulation examples of the dynamic ontology extension task. In the first example, the initial ontology is empty. Assuming the following user response:

User: *”I would like to ask how to go to the National and Kapodistrian University of Athens”*

In this example, the module generates four object properties, five datatype properties, eight new classes, and four instances.

Regarding the second example, the input ontology is the one previously created. Assuming the user utterance is as follows:

User: *“I live in Thessaloniki with my friend Maria.”*

In this case, two new instances, “Thessaloniki” and “Maria”, are added to the “Person” and “City” classes, respectively. Additionally, two new object properties, named “live” and “friend”, are added to establish relationships between the speaker and the new instances.

In the ontology extension process, when an entity is labeled as “PERSON,” a new class named “Person” is created if it does not already exist. Subsequently, for each entity labeled as “PERSON,” a new instance is added to the corresponding “Person” class. Similarly, if an entity is classified as “GPE”, after the disambiguation phase, a new class named “City” or “Country” is created if it is not already present. If a class named “City” is created and the entity’s status as a capital is verified using ISA relations from BabelNet, a subclass named “Capital” is established under the “City” class. Then, to further expand the ontology dynamically, a new class named ”Country” is introduced along with subclasses indicating the continent to which the country belongs.

In the case where the language analysis tool identifies an entity as an organization, a new class named “Organization” is generated. Moreover, a subclass is created to indicate the organization’s type, which is retrieved from hypernym relations from BabelNet. Finally, the new instance, representing the organization, is added. Furthermore, the city and country names associated with the organization are included in the ontology, and the relevant classes are created accordingly, if do not already exist.

 ISBN: 978-1-68558-108-4

V. Evaluation

A. Ontology Evaluation

1) Debugging: In order to ensure thorough evaluation, we executed the debugger for every extended ontology. This comprehensive approach allowed us to assess the output ontology and its extensions with greater scrutiny. The employment of Protégé as the initial development tool, coupled with the utilization of the 'Pellet' reasoner and the 'Debugger' plug-in, enabled us to search for any potential faults in the ontologies. Remarkably, the debugging process confirmed the absence of faults during the validation process for all extended ontologies.

2) Ontology Metrics: The structure of the output ontology, that was presented in Section IV was evaluated using OntoMetrics, an online framework designed to validate ontologies based on established metrics. The findings obtained from the analysis conducted by OntoMetrics are provided in Table III. The metrics presented in Table III encompass both simple metrics, such as the count of classes, axioms, and objects, as well as schema metrics. The simple metrics, categorized as Base Metrics, provide insights into the number of ontology elements. On the other hand, schema metrics focus on the design aspects of the ontology, reflecting its richness, width, depth, and inheritance.

TABLE III
ONTOLOGY METRICS GENERATED BY ONTOMETRICS.

Category	Metric	Value
Basic	Axioms	108
Basic	Class Count	10
Basic	Object Property Count	7
Basic	Data Property Count	6
Basic	Individual Count	6
Basic	Description Logic Expressivity	ALI(D)
Schema	Attribute richness	0.6
Schema	Inheritance richness	0.9
Schema	Relationship richness	0.4375
Schema	Axiom/class ratio	10.8
Schema	Inverse relations ratio	0.285714
Schema	Class/relation ratio	0.625

B. Application-based evaluation

It is important to acknowledge that the proposed method achieves around 98% accuracy rate in expanding the ontology for entities categorized as locations, persons, or organizations. This achievement highlights the efficacy and robustness of the proposed approach, establishing its potential as a valuable tool for ontology expansion and enhancement in domains where locations, persons, and organizations play a significant role.

Furthermore, the proposed approach successfully fulfills its objective of enhancing the agent's intelligence and significantly reducing the time required for information retrieval. This efficiency is achieved through the generation of new knowledge autonomously, eliminating the need to ask certain questions. Consequently, this reduction in interaction time not only enhances user satisfaction but also reinforces the perception of the agent's advanced intelligence. Moreover, the approach enables seamless multilingual interactions by comprehending entity names in various languages. This feature proves particularly beneficial for individuals who may lack fluency in verbal communication, thus promoting inclusivity and accessibility in human-agent interactions. By encompassing these capabilities, the proposed approach demonstrates its potential to improve user experience and facilitate effective communication across diverse linguistic backgrounds. Leveraging the geographical coordinates, the chatbot possesses the capability to determine the migrant's location accurately. Consequently, it can provide tailored recommendations of nearby places, organizations, and public sectors based on the user's specific inquiries.

In the rest of this Section, we will present a compilation of competency questions alongside the corresponding SPARQL queries and their exceptionally satisfactory outcomes.

CQ1: Given a specific city, determine the corresponding country and identify the official language spoken within that locality.

The following SPARQL query returns the country that has a city named "Thessaloniki" and the official languages of the specific country.

```
SELECT ?country ?language
WHERE {
  ?country rdf:type ex:country.
  ?country ex:has_city ex:Thessaloniki.
  ?country ex:official_languages ?language
      .}
```

Listing 1. SPARQL Query CQ1

The output (Table IV) encompasses the country names along with their corresponding official languages.

TABLE IV
RESULTS CQ1

a/a	Country	official_languages
1	Greece	"Greek"
2	Greece	"Demotic_Greek"

CQ2: Given a specific organization, determine the type of the organization in order to understand a user's background and work experience.

The following SPARQL query returns the type of a given organization.

```
SELECT ?class
WHERE {
  ex:UBS a ?class .}
```

Listing 2. SPARQL Query CQ2

Table V displays the outcomes obtained from CQ2.

CQ3: Given a specific city name in German, return the name of the city in French.

The next SPARQL query yields the French translation of the city name "Thessaloniki," despite it being stored in German.

TABLE V
RESULTS CQ2

a/a	Classes
1	owl:Thing
2	Organization
3	Bank

```
SELECT ?frenchName
WHERE {
  ?city rdf:type ex:City;
       ex:has_value "Thessaloniki"@de;
       ex:has_value ?frenchName.

  FILTER (LANG(?frenchName) = "fr")}
```

Listing 3. SPARQL Query CQ3

Table VI illustrates the output of the SPARQL query CQ3, which is the French translation of the city “Thessaloniki”.

TABLE VI
RESULTS CQ3

a/a	French Name
1	“Thessalonique”@fr

VI. CONCLUSION

In summary, this research paper presented a dynamic ontology extension approach that leverages relationships from BabelNet. The approach was implemented within an ontology-based chatbot system. The results demonstrate that with each user interaction, the chatbot continually enhances its intelligence by integrating new information into the ontology. This progressive intelligence augmentation leads to significant reductions in interaction time and heightened user satisfaction, as users perceive the chatbot’s increasing inteligence. Additionally, the proposed method enables the chatbot to engage in multilingual conversations, effectively bridging language barriers and accommodating users with diverse backgrounds.

Overall, this research highlights the potential of the dynamic ontology extension approach in creating more intelligent and adaptable chatbot systems. Future investigations could explore further advancements in utilizing BabelNet relationships and extend the multilingual capabilities to enhance user experience in a broader range of linguistic contexts.

ACKNOWLEDGEMENT

This work has received funding by the European Commission as part of its H2020 Programme under the contract number 870930-IA (WELCOME), grant agreement No. GA101017558 (ALAMEDA), and grant agreement No. GA101049294 (LEAGUE).

REFERENCES

[1] S. Decker et al., “The semantic web: The roles of XML and RDF,” IEEE Internet Computing, 4, 63-73, 2000.
[2] L. Ma et al., “Towards a complete OWL ontology benchmark,” 3rd European Semantic Web Conference, ESWC 2006 Budva, Montenegro, June 11-14, 2006, pp. 125-139, 2006.
[3] A. Hallili, “Toward an ontology-based chatbot endowed with natural language processing and generation,” 26th European Summer School In Logic, Language and Information, 2014.
[4] H. Al-Zubaide and A. Issa, “Ontbot: Ontology based chatbot,” International Symposium On Innovations In Information And Communications Technology, pp. 7-12, 2011.
[5] A. Vegesna, P. Jain, and D. Porwal, “Ontology based chatbot (for e-commerce website),” International Journal Of Computer Applications, 179, 51-55, 2018.
[6] S. Varitimiadis, K. Kotis, D. Spiliotopoulos, C. Vassilakis, and D. Margaris, “Talking triples to museum chatbots,” Culture And Computing: 8th International Conference, Held As Part Of The 22nd HCI International Conference, 2020, pp. 281-299, 2020.
[7] L. Wanner et al., “Kristina: A knowledge-based virtual conversation agent,” Advances In Practical Applications Of Cyber-Physical Multi-Agent Systems: The PAAMS Collection: 15th International Conference, Porto, Portugal, 2017, pp. 284-295, 2017.
[8] E. Kamateri et al., “Knowledge-based intelligence and strategy learning for personalised virtual assistance in the healthcare domain,” Proceedings Of Semantic Technologies For Healthcare And Accessibility Applications (SyMpATHY), 2019.
[9] G. Meditskos et al., “Towards an ontology-driven adaptive dialogue framework, Proceedings of The 1st International Workshop On Multimedia Analysis And Retrieval For Multimodal Interaction, pp. 15-20, 2016.
[10] C. Avila et al., “MediBot: an ontology based chatbot for Portuguese speakers drug’s users,” International Conference On Enterprise Information Systems, 1 pp. 25-36, 2019.
[11] R. Navigli and S. Ponzetto, “BabelNet: Building a very large multilingual semantic network,” Proceedings Of The 48th Annual Meeting Of The Association For Computational Linguistics, pp. 216-225, 2010.
[12] G. Flouris, D. Manakanatas, H. Kondylakis, D. Plexousakis, and G. Antoniou, “Ontology change: classification and survey,” The Knowledge Engineering Review, 23, 117-152, 2008.
[13] M. Asim, M. Wasim, M. Khan, W. Mahmood, and H. Abbasi, “A survey of ontology learning techniques and applications, Database, 2018.
[14] P. Cimiano and J. Völker, “A framework for ontology learning and data-driven change discovery,” Proceedings Of The 10th International Conference On Applications Of Natural Language To Information Systems (NLDB), 3513, pp. 227-238, 2005.
[15] J. Völker, S. Langa, and Y. Sure, “Supporting the Construction of Spanish Legal Ontologies with Text2Onto,” Springer, 2008
[16] E. Drymonas, K. Zervanou, and E. Petrakis, “Unsupervised ontology acquisition from plain texts: the OntoGain system,” 15th International Conference on Applications of Natural Language To Information Systems, NLDB 2010, June 23-25, 2010, pp. 277-287, 2010.
[17] M. Hajji, M. Qbadou, and K. Mansouri, “An adaptation of Text2Onto for supporting the French language,” International Journal Of Electrical and Computer Engineering (2088-8708), 10, 2020.
[18] X. Jing, G. Yan-Qiang, S. Ai-Ju, and X. Jian-Liang, “Chinese ontology learning technology based on Text2Onto,” 2012 7th International Conference On System Of Systems Engineering (SoSE).
[19] X. Jiang and A. Tan, “CRCTOL: A semantic-based domain ontology learning system,” Journal Of The American Society For Information Science And Technology, vol. 61, pp. 150-168, 2010.
[20] A. Memariani, M. Glauer, F. Neuhaus, T. Mossakowski, and J. Hastings, “Automated and explainable ontology extension based on deep learning: A case study in the chemical domain,” ArXiv Preprint ArXiv:2109.09202, 2021.
[21] S. Chithrananda, G. Grand, and B. Ramsundar, “Chemberta: Large-scale self-supervised pretraining for molecular property prediction,” ArXiv Preprint ArXiv:2010.09885, 2020.
[22] M. Su, C. Wu, and P. Shih, “Automatic ontology population using deep learning for triple extraction,” 2019 Asia-Pacific Signal And Information Processing Association Annual Summit And Conference (APSIPA ASC), pp. 262-267, 2019.