Digital Scholarship in the Humanities, 2023, **38**, 937–952
https://doi.org/10.1093/llc/fqac095
Advance access publication 4 January 2023
Full Paper

Building linked lexicography applications with LexO-server

Andrea Bellandi [1*]

[1]Istituto di Linguistica Computazionale 'A. Zampolli', CNR, Via Moruzzi 1, 56124, Pisa, Italy
*Correspondence: Andrea Bellandi, Istituto di Linguistica Computazionale 'A. Zampolli', CNR, Via Moruzzi 1, 56124, Pisa, Italy.
E-mail: andrea.bellandi@ilc.cnr.it

Abstract

The adoption of Semantic Web technologies in the lexicographic field, has been driven by the need to ensure the construction of lexical resources that are interoperable and can be shared and reused by the scientific communities. In this context, the OntoLex W3C working group proposed the *OntoLex-Lemon* model aimed at providing rich linguistic grounding for ontologies. It includes the representation of morphological and syntactic properties of lexical entries as well as their meaning with respect to an ontology or vocabulary. This article aims at presenting LexO-server, a set of REST services for the management of *OntoLex-Lemon* modeled lexical resources. LexO-server comes as a software backend providing data access and manipulation to frontend developers. The set of services are general enough to make possible the construction of applications oriented at different tasks, such as editing, linking, dictionary making, linguistic annotation, or NLP ones. As a demonstration of the versatility and the potential of LexO-server, we will present three web applications that rely on it.

1 Introduction

Generalizing what is stated in (Berners-Lee *et al.*, 2001), Semantic Web is an attempt to *describe* and *link* data (lexical[1] resources, in our case) in a manner that's meaningful to machines. The task of description is accomplished by means of the usage of formal representation languages based on subsets of the first order logic, in particular the Description Logic one (Baader *et al.*, 2017), such as the Ontology Web Language (OWL). Linked Data (LD) principles instead make the data linkable, by allowing each entity of a dataset (concepts, relations, attributes, and so on) be uniquely identified by an Internationalized Resource Identifier (IRI),[2] and be available on the Web via the Hypertext Transfer Protocol (HTTP). Furthermore, they recommend the use of the Resource Description Framework (RDF)[3] data model[4] for representing relationships among entities by means of triples structure <*subject*, *predicate*, *object*>. These technologies help to make data findable, reusable, accessible and web scale visible, according to the FAIR principles (Wilkinson *et al.*, 2016). From the one hand, the use of a formal language for the *representation* of lexicographic resources provides a common ground for representing and encoding their semantics while ensuring interoperability. Furthermore, the formal specification of ontologies allows performing automated reasoning on the data, for example to ensure logical consistency, to compute inferred closures, infer new knowledge on the basis of class taxonomies, property hierarchies, and so on. On the other hand, *data linking* opens up to federate with content from external resources, making it possible to deduce new facts across the web, for the discovery of new knowledge.

In the last years, this led to a number of community-driven activities that have fostered the adoption of linked data principles for the publication of language datasets, most importantly the Open Linguistics Working Group (Chiarcos *et al.*, 2012), and the Open Knowledge Foundation (Chiarcos *et al.*, 2011). They pursued the development of a Linked Open Data (sub-)cloud[5] of linguistic resources (Cimiano *et al.*, 2020a), representing an index of all the language datasets published on the Web following linked data principles. For developers of linguistic resources, this ecosystem can provide technological support or off-the-shelf implementations for common problems. Further, the distributed approach of the LD paradigm facilitates the distributed development of web of resources and collaboration between researchers that provide and use this data and that employ a shared set of technologies. One consequence is the emergence of interdisciplinary efforts to create large and interconnected sets of resources in linguistics and beyond (Monachini and Khan, 2018).

In this context, the W3C OntoLex community group,[6] developed the *lemon* model (later renamed *OntoLex-Lemon*) published as a W3C report,[7] for representing lexica that describe how ontological concepts were lexicalized in various languages. The model was adopted by a relevant number of projects of different nature (Costa *et al.*, 2021a; Ehrmann *et al.*, 2015, Sérasset, 2015; Eckle-Kohler *et al.*, 2015), and for a variety of applications that are not explicitly related to ontologies, like the modeling of lexicographic data (Del Gratta *et al.*, 2015; Tiberius and Declerck, 2017; Declerck *et al.*, 2017), specific lexical phenomena (Bellandi *et al.*, 2018; Piccini *et al.*, 2018) terminological data (Cimiano *et al.*, 2017; Arcan *et al.*, 2018), machine-readable dictionaries, including digitized versions of existing dictionaries (Gracia *et al.*, 2016; Klimek and Brümmer, 2015; Vulcu *et al.*, 2014; Villegas and Bel, 2015), and Wordnet conversions (McCrae *et al.*, 2014; Bond *et al.*, 2016; Declerck, 2020; Bajčetić and Declerck, 2021).

However a very few tools aiming to create such a resources have been developed: VocBench (Stellato *et al.*, 2020), LexO (Bellandi, 2021), Lemonade (Rico and Unger, 2015), and lemon source,[8] an editor for (Monnet) *lemon* based on the paradigm of semantic wikis (Frischmuth *et al.*, 2015). The common feature of the above tools is that they are full stack applications for end users. In this article, we present LexO-server, a backend of REST (Representational State Transfer) services for the management of *OntoLex-Lemon* resources, addressed mainly to front end developers for aiding and easing the development of end user lexicography-based applications. The set of services are general enough to make possible the construction of application oriented at different tasks, such as editing, linking, dictionary making, linguistic annotation, or NLP purposes.

The article is organized as follows. Section 2 presents the formal models on which LexO-server is based, and Section 3 provides a detailed description of the backend, in terms of what are the service types it exposes, and what are their functionalities. Some related works are discussed in Section 4. Three end user tools based on LexO-server are presented in Section 5. Finally, Section 6 draws some conclusions and discusses future works.

2 Lexical and conceptual model

The Semantic Web opens up the possibility to rethink how to produce new lexicographical products so that they more effectively respond to the needs of end users. Today, we can create lexicons in line with the FAIR principles, enriched with lexico-semantics and conceptual information. (Costa *et al.*, 2021b). In accordance with this view, LexO-server provides services for both lexical and conceptual levels. The morphological and semantic features of the lexemes composing the linguistic level are formally described according to the *Ontolex-Lemon* model. Instead, at the conceptual level, concepts and properties receive a structured and formal representation by means of the Simple Knowledge Organization System (SKOS).[9] LexO-server natively provides services for the management of SKOS ontologies only, but it makes possible lexical entries refer to external existent OWL ontologies.

The next two subsections aim at presenting an overview of the lexical and conceptual level, respectively.

2.1 The OntoLex-Lemon model

The *OntoLex-Lemon* model (McCrae *et al.*, 2012, McCrae *et al.*, 2017), developed by the OntoLex community group, has become a de-facto standard for representing and publishing lexical resources in the Semantic Web. It provides a rich linguistic grounding including different modules for the representation of phonetical, morphological, variational, and syntactic properties of lexical entries as well as the syntax-semantics interface, i.e. the meaning of these lexical entries with respect to conceptual knowledge. One of the main aspects characterizing the creation of the *OntoLex-Lemon* was the degree of openness in its development. The OntoLex community group started by collecting a set of relevant use cases,[10] thus abstracting general requirements[11] to be modeled. All the model issues and decisions are publicly available and accessible on the web.[12]

It is important to state that the model is agnostic to the specific linguistic data categories being used, allowing to reuse any data category (e.g. part-of-speech) together with its values. *OntoLex-Lemon* is able to incorporate such externally defined data categories by including their IRIs as a unique specification of a property, giving additional information such as the ownership which becomes accessible when dereferencing the corresponding IRI. In this work, we will refer to the LexInfo project (Buitelaar *et al.*, 2009) that can be viewed as a complementary model of *OntoLex-Lemon* which have fixed data categories.

The architecture of *OntoLex-Lemon* is divided into modules. Five of them define the main structure of the model, while the others are considered as extension ones. Each module accounts for specific aspects in the modeling of lexical information. Figure 1[13] depicts the modules implemented by LexO-server. In the following, the next two subsections are devoted to give a very brief overview of the main and the extension modules, respectively.

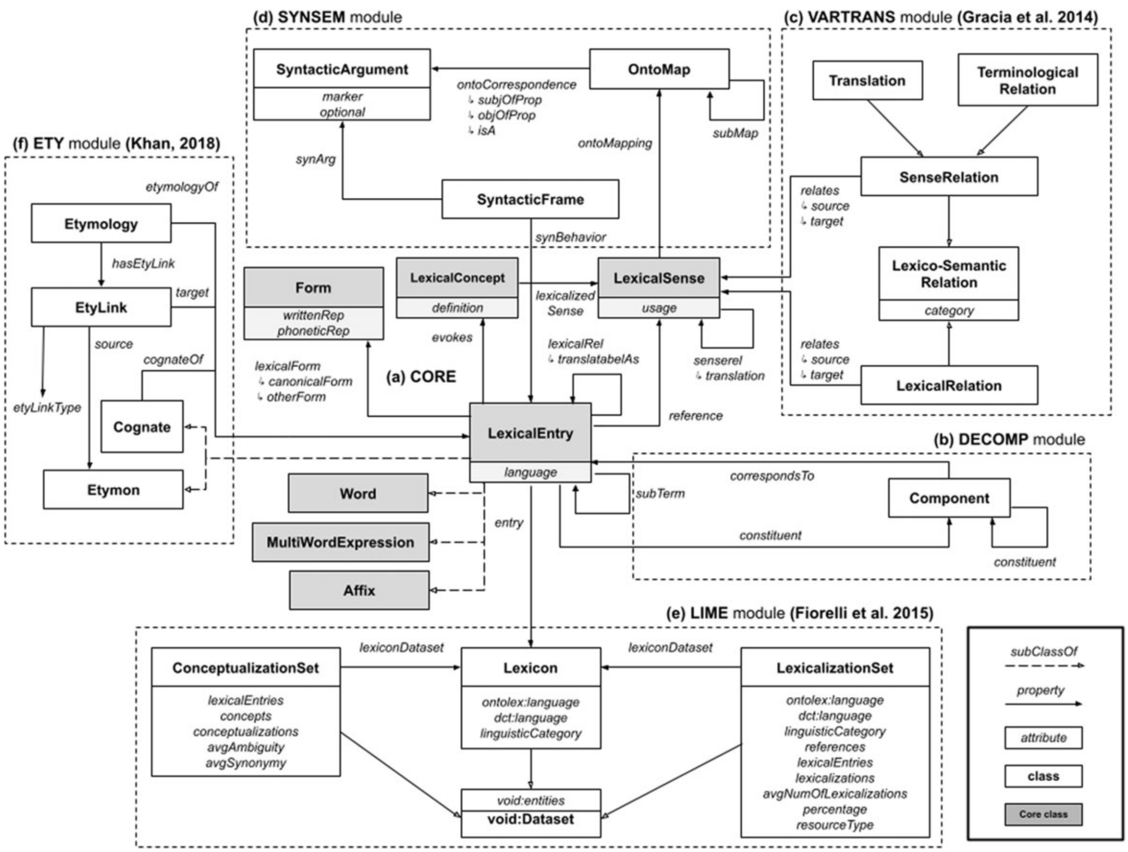

Figure 1. Overview of the linguistic model (please, refer to https://www.w3.org/2016/05/ontolex/ for a detailed overview).

2.1.1 Main modules

The main module, Figure 1(a), consists of four elements: the `ontolex: LexicalEntry` class[14] representing a set of lexical entries categorized by their types, i.e. single word (`ontolex: Word`), compound word (`ontolex: MultiWordExpression`) or part of a word (`ontolex: Affix`); the `Form` class instantiating all the inflected forms of a lexical entry, including the lemma; the `ontolex: LexicalSense` class representing a reification between the lexical entry and the concept; the `ontolex: LexicalConcept` class representing a mental abstraction, concept or unit of thought that can be lexicalized by a given collection of senses. According to the principle known as 'semantics by reference' (Buitelaar 2010), the *OntoLex-Lemon* model allows us to consider the description of morphological and syntactic behavior of a word, as separated from the ontological description of the concepts the word refers to. Those concepts can be referred to by lexical senses (`ontolex: reference` property) or denoted directly by words (`ontolex: denotes` property).

The module in Figure 1(b), is devoted to represent which are the components of a multiword. It is possible to model both the order and the morphological traits of each multiword component.

All the relations between both words and senses are modeled by the Variation and Translation module in Figure 2(c). Lexical relations, directly linking lexical entries to each other, e.g. <:LexO, lexinfo: acronym,: Lexicon_and_Ontology>, and sense relations, directly connecting lexical senses to each other, e.g., <:sense$_{tool}$, lexinfo: synonym,: sense$_{instrument}$>. The model provides the possibility to specify some properties of each relation. Indeed, the module allows to model the relation as a class. As an example, let us consider the case of translation (Gracia *et al.*, 2014). In order to represent the fact that "strumento"@it is the Italian translation for "tool"@en at a certain confidence degree, we need to create an individual `T` of the class `vartrans: Translation`, such that <:T vartrans: source strumento@en>, <:T vartrans: target tool@it>, and <:T lexinfo: confidence 0.7>.

The Syntax and Semantics module depicted in Figure 2(d) describes a syntactic frame where a given lexical item may occur by specifying its type (verb frame, transitive frame, and so on), the syntactic arguments

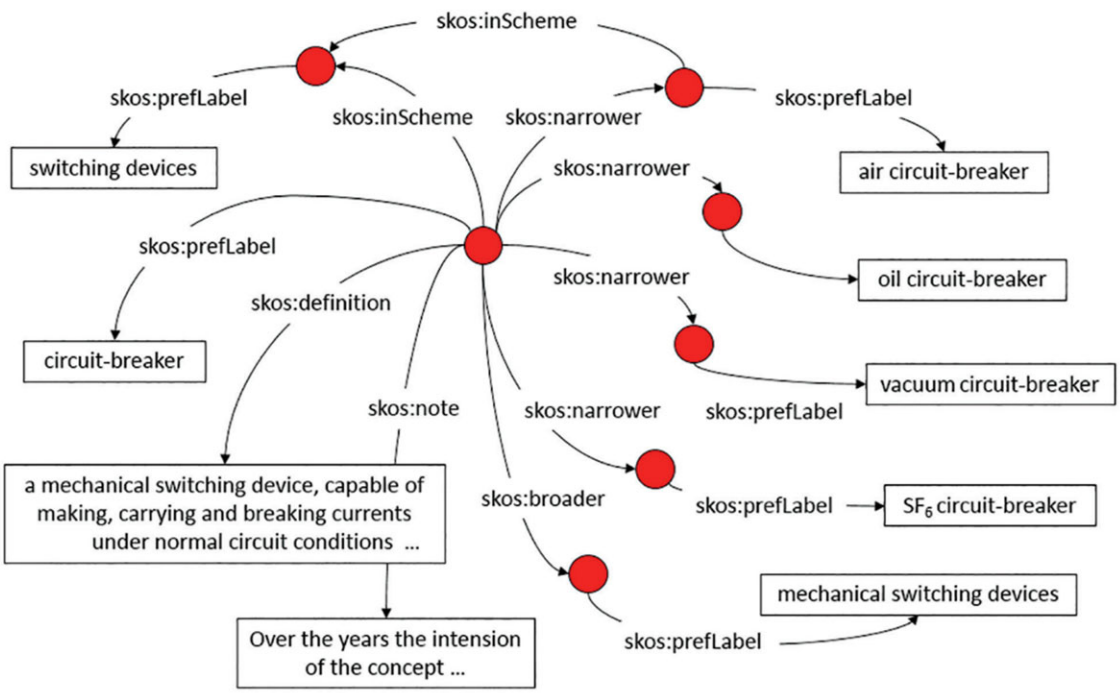

Figure 2. Graphical representation of a concept entry in SKOS (Reineke and Romary, 2019).

introduced (subject, object, etc.), their position and their mandatory or optional character in that frame. In addition, the elements of the syntactic frame can be mapped to ontological entities of the conceptual level. Let us consider for example the transitive frame 'P_1 *eats* P_2'. The predicate can be mapped to an ontological property 'D *eats* R', where the grammatical subject P_1 and the grammatical object P_2 refer to the property domain D and the property range R, respectively. D and R are ontological entities: D could represent `AnimatedEntities` and R `EdibleEntities`. More complex multi-argument structures are covered as well.

The Metadata module (Figure 2(e)), called *Lime* (Fiorelli *et al.*, 2013), provides quantitative and qualitative coarse-grained information about datasets (e.g. number of entries, of lexical senses, of conceptualizations, and so on), with the main purpose of allowing humans and machines to know what lexical material is available and thus to better understand how to use it for specific purposes.

2.1.2 Extension Modules

The modules that are currently not part of the *OntoLex-Lemon* model are:

- *lexicog*[15]: it is targeted at the representation of dictionaries and addresses structures and annotations commonly found in lexicography. This module operates in combination with the *OntoLex-Lemon* core module.
- *lemonEty*[16] (Khan, 2018a): it is aimed at representing etymological aspects of words.
- *OntoLex-Morph*[17]: it is an on-going initiative to create a vocabulary for morphologically rich languages to complement the *OntoLex-Lemon* core module.
- *FrAC*[18] *(frequency, attestation and corpus information)*: it is targeted at complementing dictionaries and other linguistic resources containing lexicographic data with a vocabulary to express (i) corpus-derived statistics (frequency and co-occurrence information, collocations), (ii) pointers from lexical resources to corpora and other collections of text (attestations), (iii) the annotation of corpora and other language resources with lexical information (lemmatization against a dictionary), and (iv) distributional semantics (collocation vectors, word embeddings, sense embeddings, concept embeddings).
- *Terminology*[19]: it is targeted at the representation of language data included in terminologies and how to relate those data to existing models for lexical data, mainly *OntoLex-Lemon* and the associated LexInfo vocabulary.

LexO-server manages the *lemonEty* module, while the implementation of *lexicog* is currently ongoing. Stable versions of OntoLex-Morph, FrAC, and Terminology, will be taken into account in the future.

The etymology module (Figure 2(f)), called *LemonEty* (Khan, 2018a), is aimed at representing the history of a word (or any other kind of linguistic phenomena) by tracing out a sort of linguistic family tree or genealogy. A candidate word is associated with several of its (postulated) etymons (`ety: Etymon` class) and cognates (`ety: Cognate` class) either directly or indirectly (indirectly in the sense of taking into explicit consideration other intervening etymons/cognates) via relations which represent historico-linguistic processes, or rather linguistic mechanisms of the sort commonly studied by historical linguists. These mechanisms are usually subsumed under one of two headings: namely, either that of borrowing, or that of inheritance (that are the possible values of the property `ety: etyLinkType`). The former refers to the process by which linguistic elements are transferred from one language into another via language contact; the latter, instead, refers to the inheritance of words (or other linguistic elements) from a parent language, or a prior stage of the same language (Khan, 2018b).

2.2 The SKOS model

SKOS (Simple Knowledge Organization System) is a vocabulary used to represent the so-called Knowledge Organization Systems (KOS), comprising taxonomies, classification schemes, and thesauri. SKOS is built upon several pre-existing Semantic Web standards for formal logic and structure such as RDF and OWL, and thus SKOS data are represented as RDF triples. The vocabulary of SKOS includes various elements among which concepts, labels, notes, relationships, and collections.

A SKOS concept (`skos: Concept`) represents any unit of thought: an idea, an object, an event; it is the single-entry node to the entry structure. A label is the element that is the descriptor in the natural language of a concept. Three label types exist: preferred label (`skos: prefLabel`) that permits to assign an authorized name to a concept, alternative label (`skos: altLabel`) that permits to assign an unauthorized name to a concept, allowing multiple same-language descriptors for a concept, and hidden label (`skos: hiddenLabel`) representing a label for performing text-based indexing and search operations, but not visible otherwise. Concepts can be organized by means of different kinds of relations that allows for the creation of a network of concepts. In particular, `skos: broader` and `skos: narrower` relations assert hierarchical relationships between concepts, i.e., that one concept is broader or narrower in meaning than another. `skos: related` relation instead, allows one to assert an associative relationship between two concepts. SKOS also offers the means of both grouping such concepts using the `skos: Collection` class, and classifying them by means of the `skos: ConceptScheme` class. Additional documentary segments are attached to the concept with a variety of relations, such as `skos: definition` that supplies a complete explanation of the intended meaning of a concept, and `skos: note` for general documentation purposes. Figure 2 illustrates a graphical representation of a typical SKOS concept entry (Reineke and Romary, 2019).

As it might be seen, SKOS and *OntoLex-Lemon* have been designed with a different purpose and use case. SKOS can only provide linguistic information by means of simple 'labels', while *OntoLex-Lemon* provides detailed information about the linguistic grounding of an ontological vocabulary, specifying in particular by which lexical entries a class or property can be verbalized. Nevertheless, they can be used in conjunction to provide more detailed information about the 'labels'. Note that SKOS allows one for attaching some metadata to a particular concept, but not to the terms defined as labels for that concept. In order to assign metadata about the labels themselves, a SKOS extension called SKOS-XL,[20] was created. It reifies the skos labels by means of the class `skosxl: Label` in order to have forms or lexical entries in the range of the `skosxl: prefLabel`, `skosxl: altLabel` and `skosxl: hiddelLabel` properties. In this way, lexical entries and forms would be inferred to be `skosxl: Labels`. However, the extension does not provide a rich linguistic grounding as the OntoLex-Lemon modules do.

The class `ontolex: LexicalConcept` introduced in Section 2.1, is represented as a SKOS concept in the *OntoLex-Lemon* model, and the property `ontolex: evokes` has to be used to relate a `skos: Concept` to an `ontolex: LexicalEntry.`[21] For example, we might want to record the actual lexical sense of a word with respect to a mental lexicon, in which 'die' evokes the event of dying, where the latter is represented as a lexical concept.

Another example of usage of SKOS in conjunction with *OntoLex-Lemon* is related to Lexical nets. They are an important type of lexical resource used very often in natural language processing applications. Lexical nets organize the senses of words into groups of equivalent meaning, so-called synsets (Miller, 1995). Further, synsets are related to each other using lexico-semantic relationships so that the resource can be regarded as a net. The Lexical model of Figure 3 presents a simple example where the depicted synset: `synTURTLE` groups all the entries that are synonyms of the term *turtle*.

The last use case we address is one where a thesaurus or other taxonomic resource or classification system in SKOS needs to be enriched with more detailed linguistic information. The conceptual model of Figure 3 reports a fragment of the animals taxonomy. The `ontolex: reference` property links all the senses

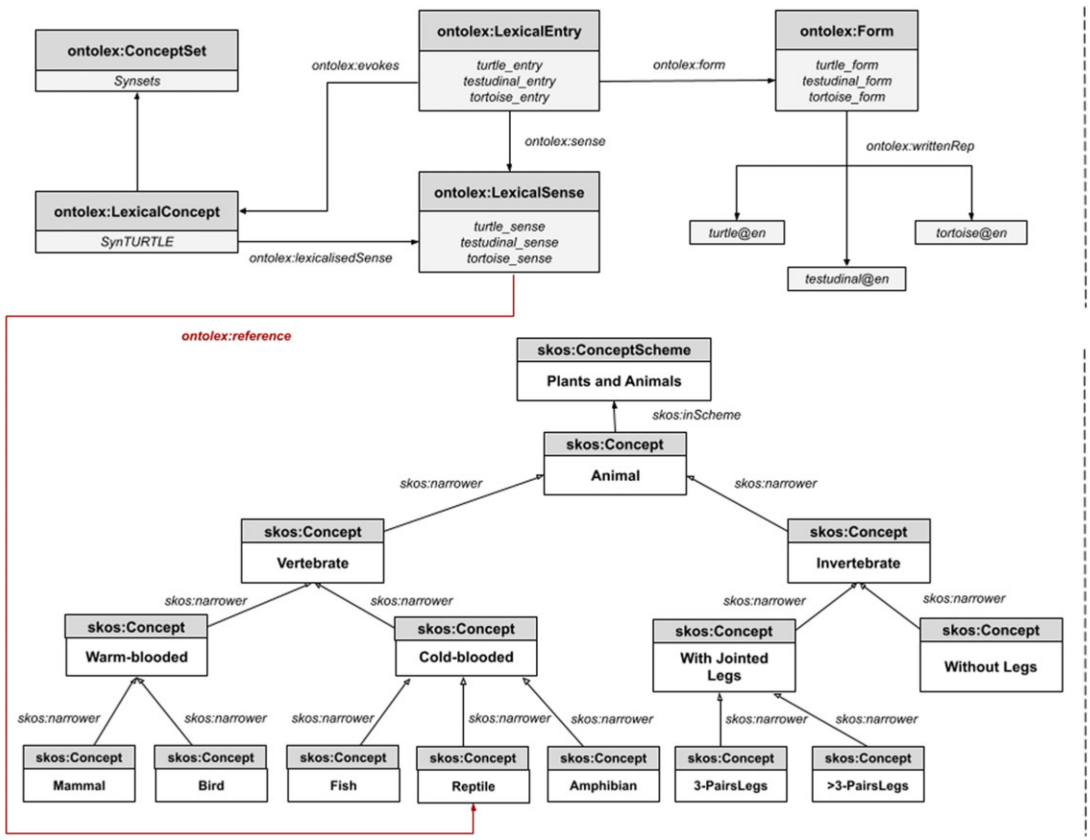

Figure 3. An example of SKOS in conjunction with OntoLex-Lemon.

grouped by the synset: `synTURTLE` to some conceptual feature in the conceptual counterpart.[22]

Note that such a representation could allow one to formulate an interesting set of queries e.g., 'The term *penguin* is a bird?', or 'How many legs does an ant have?'[23] Furthermore, an onomasiological access to terms is enabled, e.g., 'Give me all the terms indicating cold-blooded animals that are neither amphibians or fishes', or 'Give me all the terms indicating mammals or invertebrates without legs'.

3 LexO-server

In software engineering, Service-Oriented Architecture (SOA) is an architectural style where services are provided to the other components by application components, through a communication protocol over a network. A service is a discrete unit of functionality that can be accessed remotely and acted upon and updated independently, such as retrieving a lexical entry, or adding a lexical sense to a lexical entry. Service orientation is a way of thinking in terms of services and allows to maintain a strong frontend-backend separation of applications concerns in such a way that makes most services potentially reusable in different contexts. This allows developers to build different end user applications on the same backend.

LexO-server is a free and open-source backend[24] that relies on the semantic repository called GraphDB.[25] It is implemented as a set of Representational State Transfer (REST) services based on the HTTP protocol and exchanges data in JSON format.[26] Services conform to OpenAPI,[27] a specification for machine-readable interface files to describe, produce, consume, and display REST services. LexO-server has evolved from the experience of LexO-lite (Bellandi, 2021), a full stack tool for editing *OntoLex-Lemon* resources. As stated in Section 2, the LexO-server allows for managing both lexical and conceptual levels, and for a correct linking between each other.

In the next subsections, we will give an overview of the available services both for *OntoLex-Lemon* and SKOS, and we will describe some of their general features, respectively.

3.1 Services description

Concerning the lexical level, many kinds of services serving different user tasks exist. They are grouped into the following categories, based on the type of functionality they provide.

Linguistic Vocabulary. This group of services is aimed at providing the linguistic categories (LexO-server uses LexInfo as described in Section 2.1), for example grammatical categories, semantic relationships, morphological traits, types of syntactic frames, and so on. For example, when a lexical resource is being edited, user interfaces can show them as possible choices in appropriate menus.

Lexicon Creation. The services of this group allow one to create all the elements of the model presented in Section 2.1. Each lexical element has an author, represented by the user who creates it. Furthermore, each lexical entry has a state that can be equal to '*working*', meaning it is being worked on, '*completed*', meaning the description of the entry is finished, and '*validated*', meaning another user, typically a supervisor, has validated the entry. These functionalities are very useful in an editing scenario.

Lexicon Data. This group collects all the services to get both synthesis and details of resource data. Advanced searches and filters are also available. The services could enable multiple data views, for example as lists or trees, as well as detailed tables or input forms for editing purposes.

Lexicon Update. The services of this group allow modifying the elements of the model, adding or changing their properties and values.

Lexicon Deletion. The services of this group allow users to correctly delete elements or properties. Currently, the services adopt a lazy deletion policy: if an element that has to be deleted is linked to other elements of the resource, the services return a suitable error code.

Lexicon Statistics. This group of services provide some quantitative information about the resource (number of languages, number of entries per language, number of words per type, etc.) for supporting different tasks.

Graph Visualization Support. The services implement features typical of graph exploration. Currently, senses are nodes and lexico-semantic relations are edges. Some relationships may have specific characteristics, for example synonymy is symmetrical, hyperonymy is transitive, holonymy is the inverse of meronymy, and so on. Consequently, the services also specify whether the information relating to a relationship is inferred or explicit. For example, in a visualization task, an arc might be colored differently depending on the nature of the relationship it represents. Other services concern semantic distance, such as the computation of the minimum path between two senses w.r.t. a specific relation, or the distance (hops) between senses.

Query Expansion Support. The services of this group support the linguistic and semantic-based access to texts. They provide for the expansion of: (i) a word with its inflected forms and the relative morphological features, (ii) a concept with the forms of the senses the concept refers to, and (iii) a sense with the relative written forms.

Concerning the conceptual level, the services are organized in the same way as the linguistic ones, e.g., creation, update, deletion, and data. In particular, they deal with Concepts that can belong to Schemes, and be collected in ordered or unordered collections. It is possible to uniquely identify each Concept within the scope of a given concept scheme by means of the `skos: hasTopConcept` property. A set of documentation properties, is aimed at giving a definition, information about the scope of a concept, editorial information, or any other type of information. Three types of labels can be associated with a Concept: a preference label (namely, prefLabel), a hidden label (namely, hiddenLabel) and an alternative label (namely, altLabel). Label language has to be given as input to services. Finally, a specific set of services allows to manage both hierarchical and associative relationships among Concepts.

3.2 Services features

In order to simplify services (called APIs) development for users, and teams, LexO-server uses the Swagger[28] open-source tool. It helps one to design and to document APIs at scale, for easing and supporting the front-end GUI development process.

As sketched in Figure 4, each service is documented in terms of the functionality it implements, how it can be invoked, and what are its input and output formats.[29]

More specific aspects of the services can be summarized in *linking*, *federation*, and *integration*. Concerning the ability of interlinking data, LexO-server provides the possibility to link either entities of the same lexicon among each other, by performing smart searches on that entities, or external entities by assigning IRIs as properties value. For example, a synonym relation could link two lexical senses of the same lexicon, a cognate of a lexical entry could refer to an IRI representing an entity of an external published lexicon, or a lexical sense could refer to a concept defined in an external ontology (e.g., DBPedia, WikiData, and so on). Moreover, LexO-server makes available specific properties dedicated to the linking at both instance and schema level. On the instance level links can be made between individual entities (e.g., lexical entries, etymologies, multiword components, frames, and so on) using the properties *rdfs: seeAlso* and *owl: sameAs*. The first one indicates that more relevant information can be

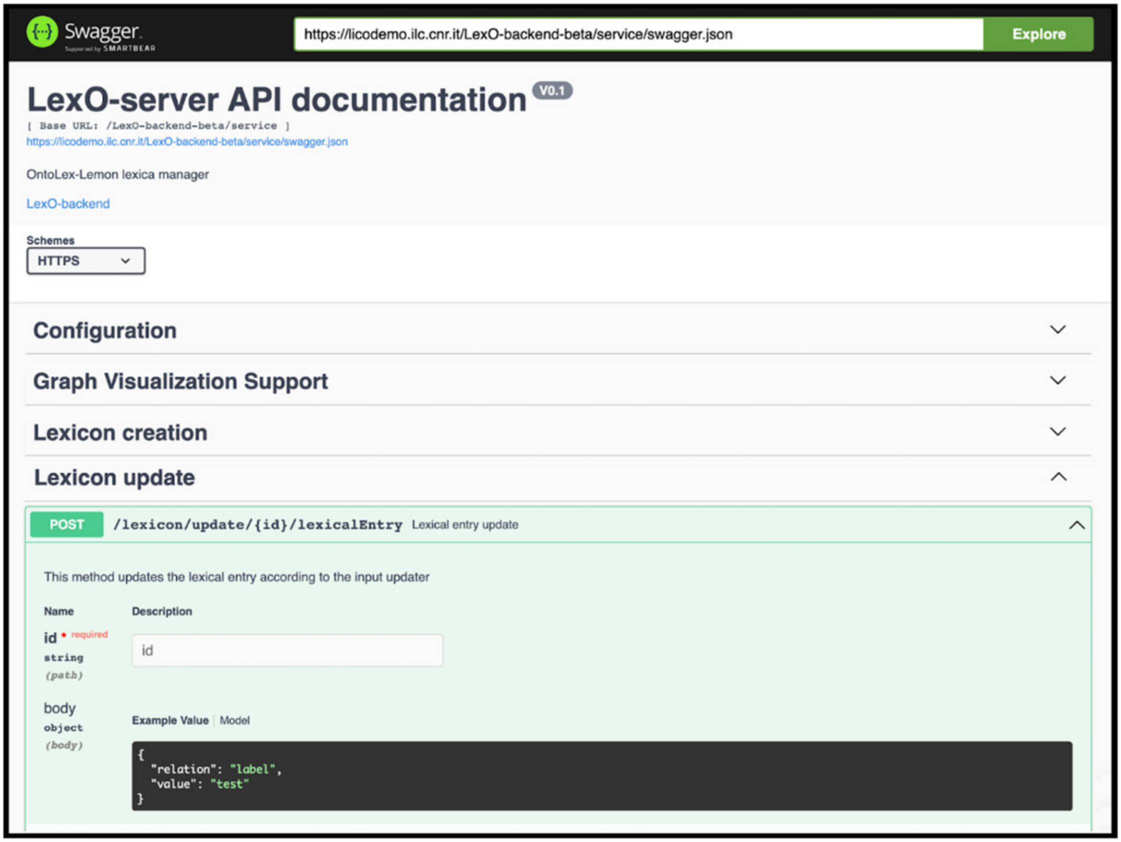

Figure 4. Swagger of LexO-server.

found by following the specific link, while the second one expresses that two IRI references actually refer to the same thing (e.g., the lexical entry one is coding and a lexical entry described in a remote dataset). On the schema level, which contains the vocabulary used to classify the instance level items, LexO-server implements the links represented by the following SKOS mapping properties used to express alignment between concepts from different vocabularies: `skos:closeMatch` for expressing that two concepts are sufficiently similar that they could possibly be used interchangeably; `skos: exactMatch` for expressing that two concepts can be used interchangeably; `skos: relatedMatch` for stating that there is an associative mapping between the two concepts.

The process of detecting links between datasets is known as link discovery. Datasets are heterogeneous in terms of their vocabularies, format and data representation. This makes the process of link discovery far more complex. Determining whether two entities from different datasets refer to the same thing is an example of what is known as the entity resolution problem. LexO-server is agnostic to any detection process and it provides a mechanism for saving the links only. However, it implements a federation system by means of which it is possible to configure a set of precompiled SPARQL[30] queries involving remote SPARQL endpoints. This enables end-user applications to automatically perform searches on external dataset for implementing advanced linking functionalities.

The implementation of the integration feature of LexO-server is currently in progress. Currently, LexO-server interacts with two specific REST services, that are Keycloak,[31] an independent server for Authentication, authorization and user management, and Zotero,[32] a tool to help one collect, organize, and cite bibliographic items. Thus, the final goal is to create a middleware that is general enough to enable LexO-server for the management of different types of external APIs for the same task. For example, it should be possible to use Zotero, Zenodo,[33] or whatever for bibliography management, Keycloak or other user management systems for access and user administration, or corpora management systems for linking a lexicon to a corpus. The integration with other services, should enable the possibility to create more complex web applications, for example collaborative and user profiled ones, or linguistic annotation ones.

LexO-server directly interacts with the REST API provided by Zotero[34] and allows for the association of bibliography information of a configurable Zotero library, to each lexical entity (lexical entry, multiword component, etymology, and so on). In particular, it is possible to specify author, title, data, pages of the bibliographic references, and to include the link to the Zotero database. *OntoLex-Lemon* does not provide any module to model bibliographic information, so LexO-server implements a system-internal data structure not yet mapped to any common ontology/vocabulary.

Authentication, authorization, and user management are handled via a direct interaction with the REST APIs of Keycloak.[35] LexO-server can be configured with the URL path and the authentication data of an installed instance of the Keycloak server. If LexO-server's services are invoked by a client sending an authentication token in the authorization header, LexO-server will validate the token against the preconfigured instance of Keycloak, and will manage the request accordingly. In absence of any authentication tokens, LexO-server will successfully manage all the requests. While the authentication process is independent of the services functionalities, authorization needs custom implementations for different use cases. Obviously, it implies that the source code of the services must be modified on the basis of the user roles chosen, and the permissions assigned to users over the data.

4 Related work

In general, there are different systems or services for the management of linguistic resources, each of which is based on different models and with different purposes for example for the construction of language lexicons, the retro-digitization of dictionaries, or the management of terminologies. To the best of our knowledge, LexO-server is novel in the context of the Semantic Web. As for the *OntoLex-Lemon* model, in fact, there are few tools dedicated to its use, and they are all full-stack applications that are aimed directly at the end users, lexicographers or terminologists. Among those ones, we cite lemon source[36], a wiki-based tool for manipulating and publishing lemon resources. It allows one to upload a lexicon and share it with other users. It is an open source project, and is freely available online for use. However, it runs older versions of the *OntoLex-Lemon* model and appears to be no longer maintained. Lemonade (Rico and Unger, 2015), is a lemon editor based on a set of lemon patterns (McCrae and Unger, 2014), a grammar that allows one to build lexical entries by expressing them in a simple user-oriented formal language, without dealing with their coding in OWL. The language, however, does not allow to specify the relationships between lexical items or between lexical senses. VocBench (Stellato *et al.*, 2017) is a web and collaborative tool for building OWL ontologies, SKOS thesauri and generic multilingual RDF datasets. Among other features, it also offers editing functionality for *OntoLex-Lemon* assets (Stellato *et al.*, 2020). Finally, LexO (Bellandi, 2021) is a system dedicated to the editing of *OntoLex-Lemon* resources with a user interface that abstracts the complexities of the model to the scholar. A detailed comparison between LexO and VocBench is discussed by the authors in (Fiorelli *et al.*, 2020).

However, there are some works that offer REST services similar to those described here.[37] A set of APIs[38] was developed by K Dictionaries[39] (formerly Kernerman Dictionaries), for the use of many bilingual dictionaries in over 50 languages, including a series of learner dictionaries of various languages. The University of Oxford has instead developed some APIs[40] that contain several monolingual English dictionaries (intended for native English speakers) and bilingual dictionaries in over 30 languages combined with English. Finally, the University of Cambridge has developed a series of APIs[41] for accessing their monolingual English dictionary for students. The recent European ELEXIS project[42] instead, outcame a collection of monolingual and multilingual resources with a broad range of usage, such as historical dictionaries and terminological resources, available for most European languages, also containing lexical resources represented by the *OntoLex-Lemon* model. A REST interface was developed for providing access to the resources.[43] To this extent, it provides a number of basic tools to offer indexing and search over the dictionary interface. As the interface is intended to be implemented with very little effort for the contributors to the ELEXIS network there is a focus on making minimal and simple queries, as such the interface only documents very basic usage. More sophisticated usage can be provided by either custom extensions or by downloading all the data and querying it offline.[44]

However, all the services cited above are exclusively oriented to the access and use of existing mono and multi-language lexical resources, while LexO-server is also focused on the offer of editing services, in order to serve a wider set of possible tasks for the development of dedicated applications by third parties. Another aspect that characterizes and distinguishes LexO-server from the services mentioned above is the ability to refer to extra-linguistic ontologies to assign a formal description to the meaning of words or directly to the lexical entries, thus allowing conceptual access to linguistic resources.

5 Practical implications

Reusability is the main feature of software components developed with a service-oriented architecture. Indeed, frontend developers can build different end-user applications based on LexO-server. Here, we are going to briefly present three web tools, in order to illustrate the versatility and the potential of LexO-server. They implement three different tasks: the editing of lexical resources, the visualization of the lexico-semantic relations of a lexicon, and the support to the linguistic-based search on texts.

Concerning the first one, the authors in (Quochi *et al.*, 2022) developed a tool called EpiLexO, a web application dedicated to the creation and editing of lexical resources for ancient fragmentary languages integrated, i.e., linked, to their 'testimonies' (i.e., transcriptions of epigraphic texts), to related bibliography, to contextual metadata, and to other relevant independent resources, such as the LiLa Knowledge Base (Mambrini *et al.*, 2020) and common vocabularies. It was conceived and realized for a project of 'Languages and Cultures of Ancient Languages', and thus tailored on the restsprachen of ancient Italy; typical target users are historical linguists. As shown in Figure 5, the column on the left of the interface shows the lexicon structure with some summary data such as the editing status of the lexical items and the user who created them. At the top, it is possible to carry out advanced searches based on some parameters. The services of the Lexicon Data group perform the above functionalities. The details of the entries selected in the left column of the interface, are reported in the center of the GUI. These can be created, modified, and deleted through the services of the Lexicon Create, Lexicon Update, and Lexicon Deletion groups. The context information of what has been selected in the left column is edited in the rightmost column, such as the bibliography, external links, notes, and some metadata.

The work presented in (Colombo and Giovannetti, 2022), is a first experimental tool for allowing the navigation of 'PAROLE-SIMPLE-CLIPS' (PSC), a computational lexicon of Italian, developed from 1996 to 2003 by the Institute of Computational Linguistics 'A. Zampolli' (Ruimy *et al.*, 2002).[45] The tool shows the lexical senses and their relationships by means of a graph. As shown in Figure 6, the interface is divided into two parts. The left column shows the list of senses available in the lexicon with the possibility of filtering them on the basis of various parameters. This part of the interface uses the services of the Lexicon Data group. In the center of the interface, two different use cases are reported: the closure computation of a semantic relation, and the navigation by a specific relation. Figure 6(a) shows the transitive closure of the synonym relation related to the synonyms of '*trabiccolo*' ('*contraption*' in English). The arrows in green are the inferred relations, since synonyms are symmetric and transitive. This kind of visualization groups a set of equivalent senses, so called synsets. Figure 6(b) shows a navigation of lexical senses by the meronym relation starting from the term '*mano*' ('*hand*' in English). The hierarchy of its meronyms is reported beneath it (for example '*finger*', '*nail*', and so on), while its holonyms are represented in the upper ('*arm*', '*body*'). The main services exploited by this application are those of the Graph Visualization Support group.

Finally, in Giovannetti *et al.* (2021), the authors developed a tool to access a text on a linguistic and conceptual basis. Linguistic level is represented by the PSC resource, while the conceptual one by the SIMPLE ontology (Lenci *et al.*, 2001), where each concept is referenced by a lexical sense of PSC. Figure 7 shows an example of linguistic-based text search. The application has been developed as part of the 'Translation Project of the Babylonian Talmud',[46] and manages with the Italian translation of the sacred text of Jewish culture. As reported in Figure 7, the user asks for contexts of the word '*recipiente*' ('*recipient*' in English). The interface proposes the ambiguous lexical entries returned by LexO-server. The user selects the desired senses having *noun* as grammatical category, and specifies that the desired forms must be in the plural form. LexO-server expands the query (by means of the services of the query expansion support group) with the direct hyponyms of the chosen forms (Figure 7 below) and sends the list of the resulting forms to the text search module. Example of retrieved text segments are: 'le *anfore* intatte vanno al fiume' (the intact *amphorae* go to the river), 'quelli sono proprietari di *barili*' (those are *barrels* owners), 'Rav Chisdà lo incoronava circondandolo di *bicchieri* di vino' (Rav Chisdà crowned him by surrounding him with *glasses* of wine). It is also possible to access the desired contexts of the text starting from conceptual (or extra-linguistic) characteristics. By composing a query relating to the senses referring to the semantic field of 'Animal of the sky' for example, the contexts relating to the forms (only lemma or even inflected forms) of the senses of 'bird', 'fly', 'grasshopper', and so on, will be returned.

Concerning computational linguistic tasks, LexO-server's services can also be exploited in NLP pipelines. We sketch some examples in the following. The access to the link between lexical and conceptual level, could give an interpretation of a word in terms of the

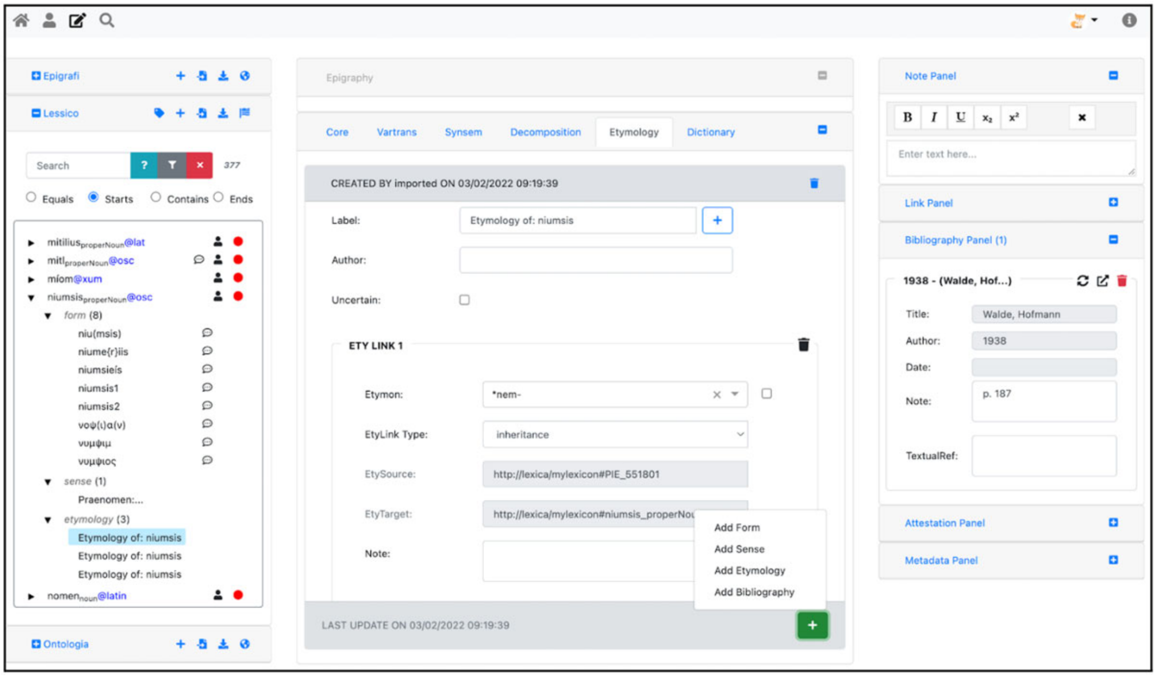

Figure 5. The EpiLexO GUI.

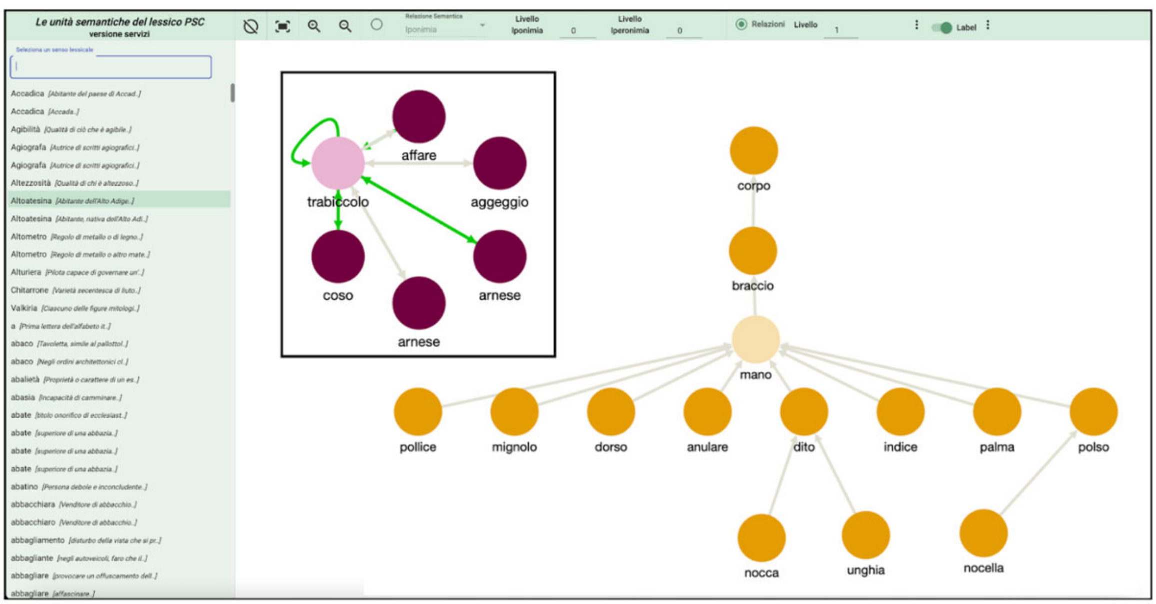

Figure 6. GUI of the lexicon graph tool.

identifiers that would be generated by the semantic parsing of a sentence. Let's consider the sentence '*Apple launched the M2 MacBook Air in 2022*'. In the context of a language understanding task, the verb 'launched (in)' may be understood as generating the IRI of the concept of launchDate.[47] Language generation and translation tasks could benefit from the services that compute semantic distance and translation equivalents, respectively. Finally, advanced access to lexical senses, provided by the services belonging to the Lexicon Data group, could support Word Sense Disambiguation, and lexical substitution tasks.

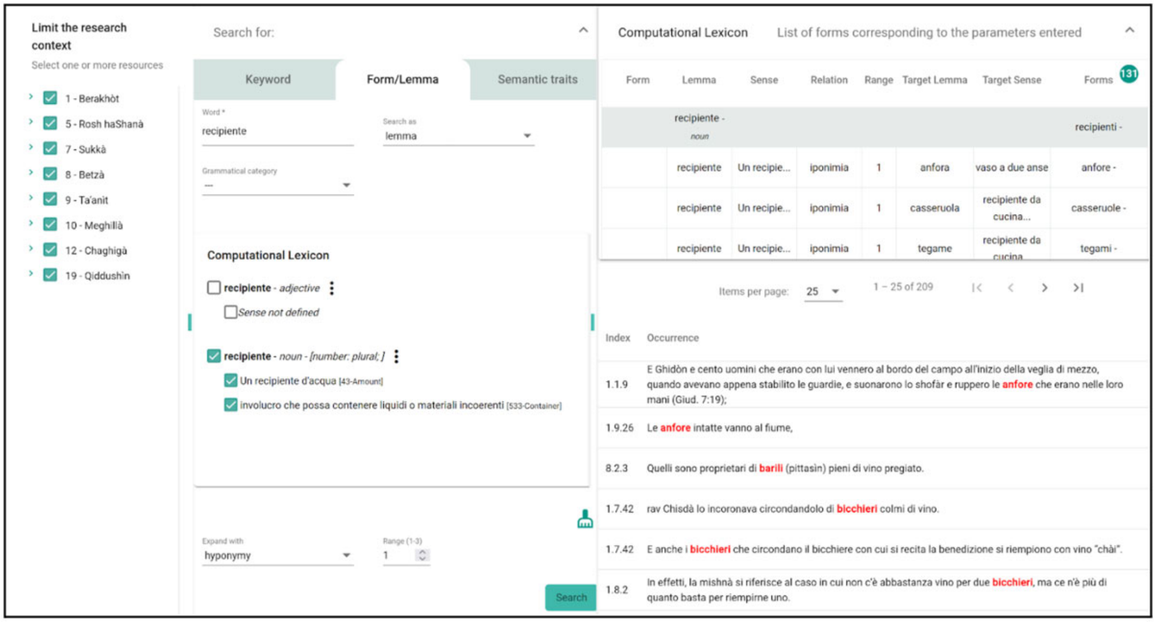

Figure 7. GUI of the linguistic-based text search system.

6 Conclusion and future work

In this article, we presented LexO-server, a set of REST services for Linguistic Linked Data, in particular for the management of *OntoLex-Lemon* modeled lexical resources. The services are general enough to make possible for implementing different lexicographic tasks such as editing, visualization, text search, or NLP ones. However, the linguistic model managed by LexO-server has been gaining wide acceptance in the Digital Humanities (DH) domain, in particular in lexicography and philology fields, as an interoperable representation formalism for electronic dictionaries (regardless of whether accompanied with an ontology). The relevance of LexO-server in this domain is two-fold. From the one hand, even if DH will continue to maintain community-specific standards, most importantly the Text Encoding Initiative (TEI), synergies between general Linguistic Linked Data vocabularies and resources and DH-specific approaches, will increase. As shown in Section 5, the management of etymology, the possibility to document information with bibliographical references, and the specification of the linguistic information confidence, make LexO-server suitable for the creation of historical lexica by philologists, historians, and linguists. On the other hand, the great heterogeneity of scientific disciplines and user communities involved in the DH, leads to heterogeneity of data formats and data sources that represents a technical challenge from the point of view of interoperability. LexO-server facilitates the integration of heterogeneous data formats and distributed data sources.

Further work goes into two directions: the first one will be dedicated to bug fixing and performance improvements, while the second one will be dedicated to implement the following new aspects, some of which are currently part of research topics:

- *linguistic categories agnosticity*. Although the most used catalog of linguistic categories with *OntoLex-Lemon* is Lexinfo, many other catalogs exist such as datCatInfo,[48] the CLARIN concept registry,[49] OLiA,[50] and GOLD.[51] Currently, LexO-server relies on Lexinfo only. We retain that a further generalization of LexO-server, could be that of the transparency w.r.t. a specific catalog of linguistic categories.
- *OWL ontologies*. OWL is obviously more expressive than SKOS. It was designed to represent rich and complex knowledge about things, groups of things, and relations between things. Its dialects or profiles underlie different decidable subsets of the first order logic. OWL allows expressing statements as logical axioms, by means of logical operators such as intersection, union, complement, and so on. The main disadvantage of using triple stores such GraphDB as LexO-server relies on, is the need of mapping an OWL ontology, that is the set of logical axioms, into an RDF graph. The report 'OWL 2

Web Ontology Language Mapping to RDF Graphs'[52] shows how to translate axioms that do not contain annotations, annotations, and axioms containing annotations. The management of this mapping w.r.t. the Creation Update and Delete (CRUD) operations on OWL ontologies, could be difficult to handle, and bugs prone. Alternative solutions like OWL-API (Horridge and Bechhofer, 2011) provide an axiom-centric view on OWL ontologies. It is an OWL native API, containing a number of OWLAxiom objects, and some convenience methods for axioms management. It provides a level of abstraction that insulates developers and applications from underlying syntactic presentations, in particular RDF or triple-based representations. Unfortunately, high-performing storing and inferencing mechanisms implementations of OWL-API, do not exist. Instead, triple stores provide very high performance about the management of very large datasets. These are the technological aspects we will consider in the development of his kind of services.

- *metadata and lexicographic information*. LexO-server shall allow for the exporting of the data in LD compliant formats. While the *OntoLex-Lemon* model is natively fully LD compliant, we still need to make decisions on how to represent provenance, bibliography, and bibliographic references or citations. The PROV-O ontology (Lebo, 2013) includes a notation standard for provenance that is easy for humans to read, methods for accessing and querying provenance, and a few other sub specifications.[53] Good candidates for bibliographical aspects are the FRBR-aligned Bibliographic Ontology (FaBiO) or the Citation Typing Ontology (CiTO) (Peroni and Shotton, 2012). However, this is a research topic in the humanities and other options still have to be taken into account.
- *non-LD models*. The Text Encoding Initiative[54] (TEI) is probably the most dominant model for the lexicographic community. It provides diverse alternatives for encoding different kinds of lexical resources, as well as for modeling the same lexical information, through the XML-based technology. Also, the representation of terminological data is based on the same technology and relies on the TermBase Exchange[55] (TBX) format (ISO-30042: 2019) which is based on the Terminological Markup Framework (TMF) structural meta-model (ISO-16642: 2017). However, TEI and TBX were not designed with the purpose of linking data on the web but rather as a means of ensuring the exchange, and consequently, the reuse of data. In this regard, numerous studies have been conducted for combining XML-native models and the Semantic Web. Concerning TEI, the most recent one is described in (Bański *et al.*, 2017), where the authors formulated TEI-Lex0, a customization of TEI schema, aimed at establishing a target format to facilitate the interoperability of heterogeneously encoded lexical resources. Concerning TBX, approaches to convert terminological data to RDF have been proposed, in order to make those data part of the linguistic-linked data ecosystem. Among these, we mention for example (Cimiano *et al.*, 2015) which proposes a conversion system based on the Ontolex-Lemon model, and a series of best practices to transform terminologies from TBX into the linked data format.[56] We will take into account these initiatives, in order to make LexO-server compliant with as many models as possible.

Funding

This work is partially financially supported by the Italian Ministry of University and Research under the PRIN 2017 programme and carried out within the 'Languages and Cultures of Ancient Italy. Historical Linguistics and Digital Models' project (PRIN 2017XJLE8J).

Author contributions

Andrea Bellandi (Methodology, Resources, Software, Supervision, Writing—original draft, Writing—review and editing).

Notes

1. From now on, the term 'data' will refer to lexical resources.
2. It builds on the Uniform Resource Identifier (URI) protocol by adding most characters from the Universal Character Set (Unicode/ISO 10646), including Chinese, Japanese, Korean, and Cyrillic characters.
3. https://www.w3.org/RDF/ (accessed 7 December 2022).
4. see also RDF Schema at https://www.w3.org/TR/rdf-schema/ (accessed 7 December 2022).
5. https://linguistic-lod.org/ (accessed 7 December 2022).
6. https://www.w3.org/community/ontolex/ (accessed 7 December 2022).
7. https://www.w3.org/2016/05/ontolex/ (accessed 7 December 2022).
8. Unfortunately, there is no publicly available version of this system that is still usable.
9. https://www.w3.org/2004/02/skos/ (accessed 7 December 2022).
10. https://www.w3.org/community/ontolex/wiki/Specification_of_Use_Cases (accessed 7 December 2022).
11. https://www.w3.org/community/ontolex/wiki/Specification_of_Requirements (accessed 7 December 2022).
12. https://lists.w3.org/Archives/Public/public-ontolex/ (accessed 7 December 2022).

13. Figure 1 does not report each class and property of the model, for lack of space. For a detailed overview of the model, please refer to https://www.w3.org/2016/05/ontolex/ (accessed 7 December 2022).
14. From now on, we will prefix each entity (classes, properties, and individuals) with the name of the ontology it belongs to followed by ":". When there is no need to specify an ontology, we will use ":" only.
15. https://www.w3.org/2019/09/lexicog/ (accessed 7 December 2022).
16. http://lari-datasets.ilc.cnr.it/lemonEty (accessed 7 December 2022).
17. https://github.com/ontolex/morph (accessed 7 December 2022).
18. https://github.com/acoli-repo/ontolex-frac (accessed 7 December 2022).
19. https://www.w3.org/community/ontolex/wiki/Terminology (accessed 7 December 2022).
20. https://www.w3.org/TR/2009/CR-skos-reference-20090317/skos-xl.html (accessed 7 December 2022).
21. It is also possible to relate a skos:Concept to an ontolex:LexicalSense by means of the ontolex:isLexicalisedSenseOf property.
22. Alternatively, it is possible to directly denote the lexical entries that are synonyms of '*turtle*' via the ontolex:denote property.
23. Since the concept the term '*ant*' refers to, could have the number of legs as property
24. https://github.com/andreabellandi/LexO-backend (accessed 7 December 2022).
25. https://www.ontotext.com/products/graphdb/ (accessed 7 December 2022).
26. https://en.wikipedia.org/wiki/JSON (accessed 7 December 2022).
27. https://www.openapis.org (accessed 7 December 2022).
28. https://swagger.io/ (accessed 7 December 2022).
29. The permanent swagger interface of LexO-server at https://licodemo.ilc.cnr.it/LexO-backend-beta/. The swagger code is embedded in the LexO-server open source project. (accessed 7 December 2022).
30. Simple Protocol and RDF Query Language is the language for querying RDF/OWL datasets.
31. https://www.keycloak.org for user management, authentication and authorization. (accessed 7 December 2022).
32. https://www.zotero.org/ (accessed 7 December 2022).
33. It is an open repository for all scholarships (https://zenodo.org/). The REST APIs of Zenodo are available at https://developers.zenodo.org/ (accessed 7 December 2022).
34. https://www.zotero.org/support/dev/web_api/v3/start (accessed 7 December 2022).
35. https://www.keycloak.org/docs-api/15.0/rest-api/index.html (accessed 7 December 2022).
36. https://lemon-model.net/download/source.php (accessed 7 December 2022).
37. A good survey is available at https://www.lexiconista.com/dictionary-apis/ (accessed 7 December 2022).
38. https://api.lexicala.com (accessed 7 December 2022).
39. https://lexicala.com/ (accessed 7 December 2022).
40. https://developer.oxforddictionaries.com (accessed 7 December 2022).
41. https://dictionary-api.cambridge.org (accessed 7 December 2022).
42. The European Lexicographic Infrastructure (https://elex.is/) (accessed 7 December 2022).
43. https://elexis-eu.github.io/elexis-rest/ (accessed 7 December 2022).
44. A recent initiative worth mentioning, originating from ELEXIS, is the NexusLinguarum COST Action (https://nexuslinguarum.eu/), that is promoting synergies across Europe between linguists, computer scientists, terminologists, and other stakeholders in industry and society, in order to investigate and extend the area of linguistic data science. It is composed of five working groups, interoperating and providing mutual feedback between themselves (accessed 7 December 2022).
45. Obviously, the tool works with whatever *OntoLex-Lemon* lexicon.
46. https://www.talmud.it/?lang=en (accessed 7 December 2022).
47. https://dbpedia.org/ontology/launchDate
48. https://datcatinfo.net/ (accessed 7 December 2022).
49. https://concepts.clarin.eu/ccr/browser/ (accessed 7 December 2022).
50. https://acoli-repo.github.io/olia/owl/ (accessed 7 December 2022).
51. http://linguistics-ontology.org/ (accessed 7 December 2022).
52. https://www.w3.org/TR/owl2-mapping-to-rdf/#Mapping_from_the_Structural_Specification_to_RDF_Graphs (accessed 7 December 2022).
53. https://www.w3.org/TR/2013/NOTE-prov-overview-20130430/ (accessed 7 December 2022).
54. https://tei-c.org/ (accessed 7 December 2022).
55. https://www.tbxinfo.net/ (accessed 7 December 2022).
56. Please, refer to (Piccini et al., 2022) for more details about the comparison of TBX and the *OntoLex-Lemon* model.

References

Arcan, M., Montiel-Ponsoda, E., McCrae, J. P., and Buitelaar, P. (2018). Automatic enrichment of terminological resources: the IATE RDF Example. In *Proceedings of the 11st International Conference on Language Resources and Evaluation*, pp. 930–37.

Baader, F., Horrocks, I., Lutz, C., and Sattler, U. (2017). *An Introduction to Description Logic*. Cambridge: Cambridge University Press.

Bajčetić, L., and Declerck, T. (2021). Interlinking Slovene language datasets. In *Proceedings of the XIX Euralex International Congress*.

Bañski, P., Bowers, J., and Erjavec, T. (2017). TEI-Lex0 guidelines for the encoding of dictionary information on written and spoken forms. In *Proceedings of the Electronic Lexicography in the 21st Century*, pp. 485–94.

Bellandi, A. (2021). LexO: an open-source system for managing OntoLex-Lemon resources. *Language Resources and Evaluation*, 55(4): 1093–126.

Bellandi, A., Giovannetti, E., and Weingart, A. (2018). Multilingual and Multiword Phenomena in a lemon Old Occitan Medico-Botanical Lexicon. *Information*, 9(3): 52.

Berners-Lee, T., Hendler, J., Lassila, O. (2001). The Semantic Web. *In Scientific American*, vol. **284**, no. 5. Scientific American, a division of Nature America, Inc., pp. 34–43.

Bond, F., Vossen, P., McCrae, J. P., and Fellbaum, C. (2016). Cili: the Collaborative Interlingual Index. In *Proceedings of the Global WordNet Conference*.

Buitelaar, P. (2010). Ontology-based semantic lexicons: mapping between terms and object descriptions. In: Huang, C.-R. *et al.* (eds), *Ontology and the Lexicon. A Natural Language Processing Perspective*. Cambridge: Cambridge University Press, pp. 212–23.

Buitelaar, P., Cimiano, P., Haase, P., and Sintek, M. (2009). Towards linguistically grounded ontologies. In *Proceedings of the European Semantic Web conference (ESWC)*, pp. 111–25.

Chiarcos, C., Hellmann, S., and Nordhoff, S. (2012) Linking linguistic resources: Examples from the open linguistics working group. In *Linked Data in Linguistics*. Berlin, Heidelberg: Springer, pp. 201–16.

Chiarcos, C., Hellmann, S., and Nordhoff, S. (2011). Towards a linguistic linked open data cloud: The Open Linguistics Working Group. *TAL*, **52**(3): 245-275.

Cimiano, P., McCrae, J. P., Rodríguez-Doncel, V., Gornostay, T, Gómez-Pérez, A, Siemoneit, B, Lagzdins, A. (2015). Linked terminology: applying linked data principles to terminological resources. In: *Proceedings of the Electronic lexicography in the 21st century, 11-13 August 2015*

Cimiano, P., Chiarcos, C., McCrae, J. P., and Gracia, J. (2020a). Linguistic linked open data cloud. In *Linguistic Linked Data*. Cham: Springer, pp. 29–41.

Cimiano, P., McCrae, J. P., Rodrìguez-Doncel, V. *et al.* (2017). Linked terminologies: applying linked data principles to terminological resources. In *Proceedings of the Electronic Lexicography in the 21st Century*, pp. 504–17.

Colombo, M., and Giovannetti, E. (2022). La Visualizzazione Grafica di Sensi e Relazioni Semantiche di un Lessico Computazionale della Lingua Italiana. In *Proceedings of AIUCD 2022*, pp. 155–60.

Costa, R., Salgado, A., Khan, A. *et al.* (2021a). MORDigital: the advent of a new lexicographical Portuguese project. In I. Kosem *et al.* (eds), *Electronic lexicography in the 21st century: post-editing lexicography. Proceedings of the eLex 2021 Conference.* Brno: Lexical Computing CZ, pp. 312–24.

Costa, R., Salgado, A., and Almeida, B. (2021b). SKOS as a key element for linking lexicography to digital humanities. In Golub K., and Liu, Y. (eds), *Information and Knowledge Organization*. Routledge.

Declerck, T. (2020). Towards an extension of the linking of the Open Dutch WordNet with Dutch lexicographic resources. In *Proceedings of the 2020 Globalex Workshop on Linked Lexicography*, pp. 33–35.

Declerck, T., Tiberius, C., and Wandl-Vogt, E. (2017). Encoding lexicographic data in lemon: lessons learned. In *Proceedings of the LDK workshops: OntoLex, TIAD and Challenges for Wordnet*, pp. 1–10.

Del Gratta, R., Frontini, F., Khan, F., and Monachini, M. (2015). Converting the parole simple clips lexicon into RDF with lemon. *Semantic Web* **6**(5): 387–92.

Eckle-Kohler, J., McCrae, J., and Chiarcos, C. (2015). LemonUby — a large, interlinked, syntactically-rich lexical resource for ontologies. *Semantic Web* **6**(4): 371–78.

Ehrmann, M., Cecconi, F., Vannella, D., McCrae, J. P., Cimiano, P., and Navigli, R. (2015). A multilingual semantic network as linked data: lemon-Babelnet. In *Proceedings of the Linked Data in Linguistics: Multilingual Knowledge Resources and Natural Language Processing*, pp. 72–81.

Fiorelli, M., Stellato, A., Lorenzetti, T., *et al.* (2020). Editing OntoLex-Lemon in VocBench 3. In *Proceedings of the 12th Conference on Language Resources and Evaluation*, Marseille, 11–16 May 2020, pp. 7196–205.

Fiorelli, M., Pazienza, M. T., and Stellato, A. (2013). LIME: towards a metadata module for ontolex. In *2nd Workshop on Linked Data in Linguistics: Representing and Linking Lexicons, Terminologies and Other Language Data*. Pisa, Italy.

Frischmuth, P., Martin, M., Tramp, S., Riechert, T., and Auer, S. (2015). OntoWiki – an authoring, publication and visualisation interface for the Data Web. *Semantic Web*, **6**(3): 215–40.

Giovannetti, E., Albanesi, D., Bellandi, A., Marchi, S., Papini, M., and Sciolette, F. (2021). The role of a computational lexicon for query expansion in fulltext search. In *Proceedings of the 8th Italian Conference on Computational Linguistics*, Milan, June 29–July 1 2022.

Gracia, J., Villegas, M., Gómez-Pérez, A., and Bel, N. (2016). The Apertium bilingual dictionaries on the web of data. *Semantic Web Journal*, **9**(2): 231–40.

Gracia, J., Montiel-Ponsoda, E., Vila-Suero, D., and Aguado-de-Cea, G. (2014) Enabling language resources to expose translations as linked data on the web. In *Proceedings of 9th Language Resources and Evaluation Conference*, pp. 409–413.

Horridge, M. and Bechhofer, S. (2011). The OWL API: A Java API for OWL ontologies. *Semantic Web*, **2**(1): 11–21.

Khan, A. F. (2018a). Towards the representation of etymological data on the semantic web. *Information*, **9**(12): 304.

Khan, A. F. (2018b). Towards the representation of etymological and diachronic lexical data on the semantic web. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA). Miyazaki, Japan.

Klimek, B. and Brümmer, M. (2015). Enhancing lexicography with semantic language databases. *Kernerman Dictionary News*, 23: 5–10.

Lebo, T., Sahoo, S., McGuinness, D., *et al.* (2013). PROV-O: The PROV Ontology. (W3C Recommendation). World Wide Web Consortium. http://www.w3.org/TR/2013/REC-prov-o-20130430/ (accessed 28 December 2022)

Lenci, A., Bel, N., Busa, F., *et al.* (2001). SIMPLE: a general framework for the development of multilingual lexicons. *International Journal of Lexicography*, **13**(4): 249–263.

Mambrini, F., Cecchini, F. M., Franzini, G., Litta, E., Passarotti, M. C., and Ruffolo, P. (2020). LiLa: linking Latin. Risorse linguistiche per il latino nel semantic web. *Umanistica Digitale*, **4**(8): 63–78

McCrae, J. P., Bosque-Gil, J., Gracia, J., Buitelaar, P., and Cimiano, P. (2017). The Ontolex-Lemon model:

development and applications. In *Proceedings of the Electronic Lexicography of the 21st Century*, pp. 19–21.
McCrae, J. P. and Unger, C. (2014). Design patterns for engineering the Ontology-Lexicon interface. In Buitelaar, P. and Cimiano, P. (eds), *Proceedings of the Towards the Multilingual Semantic Web*, Springer Berlin Heidelberg, pp. 15–30.
McCrae, J. P., Fellbaum, C., and Cimiano, P. (2014). Publishing and linking WordNet using RDF and lemon. In *Proceedings of the 3rd Workshop on Linked Data in Linguistics*, Reykjavik, Iceland, 27 May 2014, pp. 13–16.
McCrae, J. P., de Cea, G. A., Buitelaar, P., *et al.* (2012). Interchanging lexical resources on the semantic web. In *Proceedings of the Language Resources and Evaluation Conference*, **46**(6): 701–09.
Miller, A. G. (1995). WordNet: a lexical database for English. *Communications ACM*, **38**(11) :39–41.
Monachini, M. and Khan, A. F. (2018). Towards the construction of a lexical data and technology ecosystem: the experience of ILC-CNR. In *Proceedings of the Language Resources and Evaluation Conference, Workshop Globalex*, pp. 52–54.
Peroni, S. and Shotton, D. (2012). FaBiO and CiTO: ontologies for describing bibliographic resources and citations. *Journal of Web Semantics*, **17**: 33–43.
Piccini, S., Vezzani F., and Bellandi, A. (2022). Entre TBX et Ontolex-Lemon: quelles nouvelles perspectives en terminologie? In *Proceedings of 1st International Conference on Multilingual digital terminology today. Design, representation formats and management systems*. 16–17 June 2022, Padova, Italy.
Piccini, S., Bellandi, A., and Giovannetti, E. (2018). A semantic web approach to Modelling and building a bilingual Chinese-Italian termino-ontological resource. *Book of Abstracts of the 18th Euralex International Congress*. 17–21 July 2018, Ljubljana, Slovenia, pp. 87–91.
Quochi, V., Bellandi, A., Khan, F., *et al.* (2022). From Inscriptions to Lexica and back: a platform for editing and linking the languages of ancient Italy. In *Proceedings of the Workshop on Language Technologies for Historical and Ancient Languages* (LT4HALA).
Reineke, D. and Romary, L. (2019). Bridging the gap between SKOS and TBX. *Edition-Die Fachzeitschrift für Terminologie*, **19**(2):19–27.
Rico, M. and Unger, C. (2015). Lemonade: a web assistant for creating and debugging ontology Lexica. In *Natural Language Processing and Information Systems*, Springer, Cham. Vol. **9103**, pp. 448–52.
Ruimy, N., Monachini, M., Distante, R., *et al.* (2002). Clips, a multi-level Italian computational Lexicon: a glimpse to data. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation*.
Sérasset, G. (2015). Dbnary: Wiktionary as a Lemon-based multilingual Lexical resource in RDF. *Semantic Web*, (**6**): 355–61.
Stellato, A., Fiorelli, M., Turbati, A., *et al.* (2020). VocBench 3: a collaborative semantic web editor for ontologies, thesauri and lexicons. *Semantic Web*, **11**(5): 855–81.
Stellato, A., Turbati, A., Fiorelli, M., *et al.* (2017). Towards VocBench 3: pushing collaborative development of thesauri and ontologies further beyond. In *Proceedings of the 17th European Networked Knowledge Organisation Systems Workshop*, Vol. 1937, pp. 39–52.
Tiberius, C. and Declerck, T. (2017). A lemon model for the ANW dictionary. In *Proceedings of the Electronic Lexicography in the 21st Century*, pp. 237–51.
Villegas, M. and Bel, N. (2015). PAROLE/SIMPLE lemon ontology and lexicons. *Semantic Web Journal*, **6**(4): 363–69.
Vulcu, G., Buitelaar, P., Negi, S., *et al.* (2014). Generating linked-data based domain-specific sentiment lexicons from legacy language and semantic resources. In *Proceedings of the 5th International Workshop on Emotion, Social Signals, Sentiment and Linked Open Data*, pp. 6–9.
Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., *et al.* (2016). The FAIR guiding principles for scientific data management and stewardship. *Scientific Data*, 3.