Umanistica Digitale – ISSN: 2532-8816 – n. 18, 2024
E. Giovannetti, D. Albanesi, A. Bellandi, S. Marchi, M. Papini, F. Sciolette – *Maia: an Open Collaborative platform for Text Annotation*
DOI: http://doi.org/10.6092/issn.2532-8816/19705

Maia: an Open Collaborative Platform for Text Annotation, E-Lexicography, and Lexical Linking

Emiliano Giovannetti

Cnr-Istituto di Linguistica Computazionale "Antonio Zampolli"
emiliano.giovannetti@ilc.cnr.it

Davide Albanesi

Cnr-Istituto di Linguistica Computazionale "Antonio Zampolli"
davide.albanesi@ilc.cnr.it

Andrea Bellandi

Cnr-Istituto di Linguistica Computazionale "Antonio Zampolli"
andrea.bellandi@ilc.cnr.it

Simone Marchi

Cnr-Istituto di Linguistica Computazionale "Antonio Zampolli"
simone.marchi@ilc.cnr.it

Mafalda Papini

Cnr-Istituto di Linguistica Computazionale "Antonio Zampolli"
mafalda.papini@ilc.cnr.it

Flavia Sciolette

Cnr-Istituto di Linguistica Computazionale "Antonio Zampolli"
flavia.sciolette@ilc.cnr.it

Abstract

Although open tools for manual text annotation and the creation of lexical resources have been available for some years, there is currently no integrated tool that allows, within the same environment, annotating a text corpus, building a computational lexicon, and linking linguistic annotations to lexical elements. For this reason, we have developed Maia, an open and collaborative web tool based on semantic web and linked open data technologies for text annotation, e-lexicography, and lexical linking, primarily designed and developed for use by digital humanists. This article presents the first version of Maia, describing its functionality, and outlining its software architecture and development prospects.

Keywords: text annotation, e-lexicography, lexical linking, computational lexicons, linguistic resources, Linguistic Linked Open Data, OntoLex-Lemon, Maia.

Sebbene da alcuni anni siano disponibili strumenti "open" per l'annotazione manuale dei testi e la creazione di risorse lessicali, ad oggi non esiste uno strumento integrato che permetta, all'interno del medesimo ambiente, di annotare un corpus testuale, costruire un lessico computazionale e collegare le annotazioni linguistiche agli elementi lessicali. Per questo motivo, abbiamo sviluppato Maia, uno strumento web aperto e collaborativo basato sulle tecnologie del web semantico e dei linked open data per l'annotazione di testi, la lessicografia digitale e il "lexical linking", progettato e sviluppato principalmente ad uso degli umanisti digitali. Questo articolo presenta la prima versione di Maia, descrivendone le funzionalità e delineandone l'architettura software e le prospettive di sviluppo.

Parole chiave: annotazione del testo, lessicografia digitale, lexical linking, lessici computazionali, risorse linguistiche, Linguistic Linked Open Data, OntoLex-Lemon, Maia.

1. Introduction

Text annotation and lexicon construction can be powerful allies for digital humanists: the possibility of combining the activity of annotating text and the construction of lexical resources in a synergetic action can open up unprecedented research perspectives. From the point of view of textual analysis, annotations act like digital breadcrumbs, allowing researchers to mark up specific elements (structural, thematic, semantics) within a text. This empowers them to build complex relationships between these elements, leading to richer analyses of the text's meaning and structure. At the same time, on the lexicon construction front, by identifying and grouping frequently occurring words or phrases, researchers can create custom dictionaries specific to a time period, genre, or author. This unveils unique patterns in language use, offering insights into cultural trends, social attitudes, and even individual writing styles.

Together, text annotation and lexicon construction become more than the sum of their parts. Annotated texts become training grounds for building robust lexicons, which in turn can be used to refine annotations across a wider range of texts. This cyclical process fosters a deeper understanding of the historical, social, and cultural contexts reflected in language.

In this paper we introduce Maia, an integrated tool that allows, within the same environment, corpus annotation, lexical resources construction, and the linking of annotations with elements belonging to the lexicon. While Maia is designed to annotate a text with arbitrary information, special attention has been given to annotations that involve assigning references to linguistic entities described within a computational lexicon ([16]).

Just as the task of "entity linking"[1] involves annotating a portion of text with a reference to an entity belonging to a knowledge base, or the task of "semantic annotation" ([25]), sometimes also referred to as "lexical annotation" ([4]), involves assigning to a word a reference to a meaning (usually a synset of WordNet[2] or BabelNet[3]), what we define as "lexical linking" in this context is linking words in the text to linguistic entities (lemmas, forms, senses, etc.) encoded within a computational lexicon of reference.

The motivation that led to the realisation of Maia stems from the numerous requests received by the authors of this paper in the context of various digital humanities projects to develop an integrated environment combining text and lexicon. In fact, as described in Section 2, the currently available tools tend to focus either only on annotation or on the construction of lexical resources.

Maia is intended to support a scholar (or a team of scholars, in a collaborative way) in the construction of linguistic resources from scratch (annotated corpora, lexicons, corpora linked to lexicons), as well as for the manual review of resources produced automatically by algorithms, such as linguistically analysed texts with NLP systems or lexical resources (usually terminologies or underserved language thesauri) built semi-automatically from texts ([15]), ([30]). All the details about the functionalities of Maia will be provided in Section 3, where they will be introduced, step by step, with the help of some examples.

From a more technological perspective, the tool was developed to be flexible, extensible and modular, and this criterion was reflected both in the design of its software architecture and in the choice of reference models for the representation of linguistic data. As described in Section 4, the chosen reference paradigm for data representation is Linked Data ([5]), which is inherently based on the reuse of existing data and vocabularies and lends itself naturally to resource sharing. Another criterion that guided the development of Maia was adherence to "openness" policies, both in the selection of software components used and in the open licences under which the tool was released. Maia's code and the relative documentation can be found on GitHub.[4] A demo of Maia has been made available online.[5]

Although in this article Maia is presented as a system for text annotation, lexicons construction and lexical linking, further developments are already underway that include the addition of a module for ontology construction (through which lexical senses can be linked to formal concepts) and other extensions, as summarised in Section 5.

2. Related Works

Many tools are available to support the construction of linguistic resources, but, to the best of our knowledge, there is a lack of instruments that integrate, in a unified environment, text annotation and the management of lexical and terminological data. In this section, we examine

[1] https://en.wikipedia.org/wiki/Entity_linking

[2] https://wordnet.princeton.edu/

[3] https://babelnet.org/

[4] https://github.com/klab-ilc-cnr/Maia

[5] https://klab.ilc.cnr.it/maia-demo (user: demo, password: demoversion2024)

some of the existing applications. A brief comparison between Maia and each of the tools mentioned is also provided. For this review, we considered only free and open-source applications.

2.1. Tools for text annotation

The first software we examined is INCEpTION ([14]),[6] one of the most widely used tools for text annotation. INCEpTION inspired Maia for many text annotation functionalities, in particular for its structure based on "layers", "features", and custom tagsets. INCEpTION allows the import of a "Knowledge Base" or the creation of one within the tool; this feature is envisaged in future developments of Maia (cf. Section 5), however, being more focused on the representation and management of linguistic data, Maia allows the creation of a lexical resource within it whose elements can be linked to the text by means of annotations (cf. 3.3). For corpus management, INCEpTION allows the import of texts in various formats (including CoNLL and PDF) whereas Maia currently handles "plain" texts which, however, can be marked up internally and organised into subcorpora (cf. ibid.).

Another popular annotation tool is GATE Teamware ([27]), belonging to the GATE toolkit.[7] It integrates with the language processing and information extraction applications of the toolkit itself. Teamware allows linking to ontological resources within the same environment, which, however, cannot be edited. Similar to INCEpTION, Teamware handles the corpus as a flat list of documents and does not consider an internal text structure. Annotations are based on "XML schemata" that must be loaded as XML files: Maia, on the other hand, allows the creation and description of complex annotation layers directly within the tool.

In the same category of tools, we mention Doccano,[8] which is also multi-user and multi-role. Doccano allows the import of plain texts, JSON and ConLL formats. Even in the case of Doccano, however, there is no provision for defining an internal text structure or organising the corpus into subcorpora. Unlike INCEpTION and Teamware, Doccano does not allow the definition of complex annotation levels, but uses "labels". For this reason, the possibility of linking annotations to internal or external resources is completely lacking.

Another tool worth mentioning is CophiEditor ([28]). Like Maia, it is a collaborative, multi-user, multi-role, web-based tool with a customizable layout. It is oriented towards tasks related to the creation of digital scholarly editions, allowing for the editing of texts (with editorial consistency checks), the creation of a critical apparatus, and TEI/XML export. Regarding annotation, CophiEditor is based on the principle of "Domain Specific Languages" ([29]).

The last tool we mention is CATMA,[9] conceived by its authors "to emulate the flexible workflows of hermeneutic text interpretation". In CATMA, annotations are organised by "tags", which can be structured in hierarchical "tagsets". However, unlike the way Maia describes layers through "features", it is not possible to equip annotation tags with distinctive attributes that go beyond the mere label, just as is the case in Doccano.

[6] https://inception-project.github.io/

[7] https://gate.ac.uk/download/#latest

[8] https://github.com/doccano/doccano

[9] https://catma.de/

2.2. Tools for lexicons creation

Focusing on applications for lexicon editing, we firstly mention LexO ([2]),[10] a collaborative web-based editor based on the OntoLex-Lemon model (cf. Section 4.1.1). A substantial part of Maia's lexicon editing interface is inspired by LexO (cf. 3.1) and the services for storing and processing lexicon data are based on the LexO-server module (cf. Section 4.1.2).

To the best of our knowledge, the only other two open-source tools available for the creation of lexicons in the Linked Data framework are VocBench ([24])[11] and LiFE ([23]). VocBench is a collaborative web editor for RDF resources. Unlike Maia, which is based on a specific model and is designed for users who are not experts in semantic web technologies, VocBench places the responsibility for the correct use of the adopted vocabulary classes and properties on the user, in order to provide maximum freedom of choice and action.

LiFE[12] is a more recent tool, conceived for the creation of a unified environment for managing linguistic data collected from the field (lexicon, interlinear glossed text[13] and associated multimedia content) to create lexicons based on the OntoLex-Lemon model. It also provides a pipeline for the automatic extraction (and contextual POS-tagging) of snippets of texts.

In the context of field investigations in linguistics, a widely used tool is FieldWorks Language Explorer (FLEx)[14] for the organisation and analysis of linguistic data, represented within it in XML. Unlike Maia, it does not follow a Linked Open Data-oriented approach, but it does allow for the management of lexical data and its association with texts. In FLEx, the lexicon is managed as a set of entries ordered in tables that list form, lemma, gloss, and grammatical category, which can be converted into a dictionary entry view. For corpus management, unlike Maia, FLEx automatically annotates texts entered into the system based on the lemmas present in the lexicon and their related information, such as POS, glosses, and morphemes.

Furthermore, defined as a "Dictionary Writing System", Lexonomy ([17])[15] is a tool developed as part of the Horizon 2020 ELEXIS infrastructure for lexicography.

This tool is based on the TEI-Lex-0 model,[16] in a format compliant with OntoLex-Lemon. Lexonomy, for which an online instance is available upon registration, allows for the creation of monolingual and bilingual dictionaries. In this sense, Lexonomy differs from the previously

[10] https://github.com/andreabellandi/LexO-lite

[11] https://vocbench.uniroma2.it/

[12] For the alpha version, cf. https://github.com/unrealtecellp/life. A video demonstration is here: https://www.youtube.com/watch?v=HJWCjeiv3mU

[13] Interlinear glossed text represents a type of notation frequently used in linguistic research, characterised by the alignment between morphemes and corresponding grammatical values, cf. ([6])

[14] https://software.sil.org/fieldworks

[15] https://www.lexonomy.eu/

[16] TEI-Lex-0 ([21]) is technical specification and a set of community-based recommendations for encoding machine-readable dictionaries, with a mapping on OntoLex-Lemon (https://github.com/elexis-eu/tei2ontolex), and developed in conjunction with the revision of the ISO LMF standard ([20]).

mentioned tools, as it is designed to create dictionaries rather than lexicons. Maia, as will be illustrated in Section 3.1, allows for the construction of both lexicons and dictionaries.

3. Maia: what it is and how it works

At its current state, Maia is a collaborative web tool that allows the creation of a lexical resource (also represented in the form of a dictionary), the constitution of a textual corpus, the multi-layered annotation of text, the visualisation of KWIC concordances, and the linking of annotations to lexical entities.

The collaborative nature of Maia is implemented first and foremost through multi-user access. New users can be added from a dedicated administration panel, and each user can define one or more "workspaces", where they can organise the layout of the various windows and the management of the various resources to their liking. Workspaces can be created in a dedicated panel that displays the list of existing workspaces, with the typical CRUD functionalities exposed.[17]

A workspace is a modular and flexible environment in which the functionalities available from the menu are displayed following the same policy as desktop windows on personal computers. Each panel can be positioned and resized according to the needs of users, and the chosen configuration will be saved upon exiting the workspace to be presented again upon the next access. Such a solution has been chosen to cover the widest possible range of use cases, allowing work on multiple resources in parallel and from different perspectives. In the remainder of this section, as the various functionalities of Maia are introduced and described, an example of workspace construction will be shown.

In general, panels used in Maia fall into two categories: **exploratory panels**, handling CRUD and navigation functionalities, and therefore appearing in a single instance per workspace; **detail panels**, expanding individual resources (whether texts, dictionary or lexical entries) and allowing their consultation and processing. In this case, it is possible to have multiple detail panels of the same type as long as they refer to different resources (for example, to parallelize the annotation of two texts).

In this chapter, the data for exemplifying the functionality of the system are taken from the "pilot" project *VocaBO* - "Vocabolario di Boccaccio Online",[18] which aims to describe the lexicon of Giovanni Boccaccio's vernacular works starting with the *Decameron*.

3.1. Lexicon and Lexicographic Resource (Dictionary)

Maia allows for the creation of both a computational lexicon (Lexicon), based on the OntoLex-Lemon data model (cf. Section 4.1.1), and a dictionary (Lexicographic Resource), based on the data model of OntoLex-Lemon's lexicographic module (ibid.). There is a close dependency between these two objects, the Lexicon and the Lexicographic Resource, which

[17] CRUD stands for Create, Read, Update, Delete

[18] For a short presentation of VocaBO project, cf. https://www.treccani.it/magazine/lingua_italiana/speciali/VocaBO/2_Pani.html

can be briefly explained as follows. The Lexicon consists of a set of Lexical Entry objects that refer to their respective Forms and Lexical Senses; the Lexicographic Resource, on the other hand, is composed of Entry (or lexicographic entry) objects that point to the Lexical Entry (which can be seen as "lemmas" in the context of a dictionary) defined by the Lexicon. Thus, it is clear that the elements of the Lexicon represent the components of the Lexicographic Resource (Figure 1). These two objects will be explained in detail below, starting with the Lexicon.

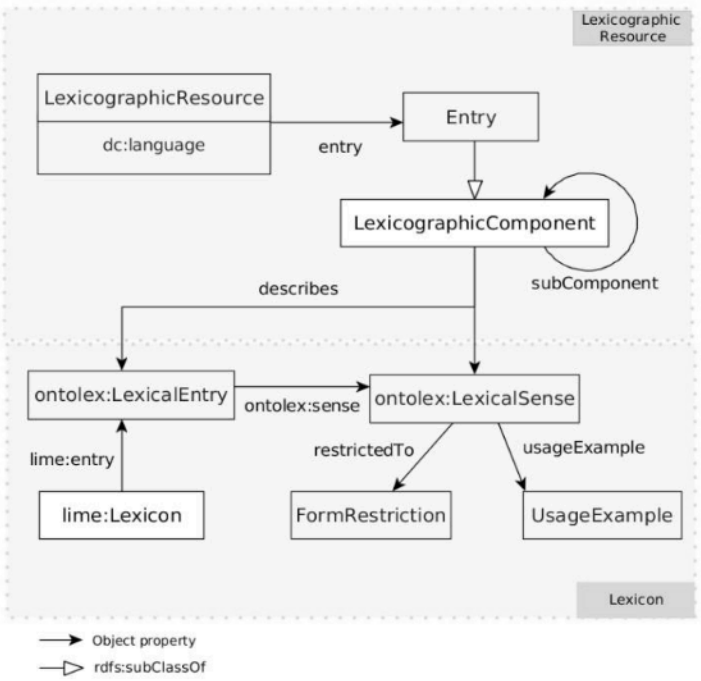

Figure 1: The OntoLex-Lemon Lexicographic data model.
(source: https://www.w3.org/2019/09/lexicog/#lexicography-module-lexicog)

3.1.1. Lexicon

Within a workspace, the Lexicon is managed through two panels: an exploratory panel, "Lexicon Explorer", and a detailed panel, "Lexicon Editor", for the description of lexical entries. The "Lexicon Explorer" panel provides the user with typical lexicon-related CRUD operations. When creating a lexical entry, the system guides the user in entering a minimum set of information.

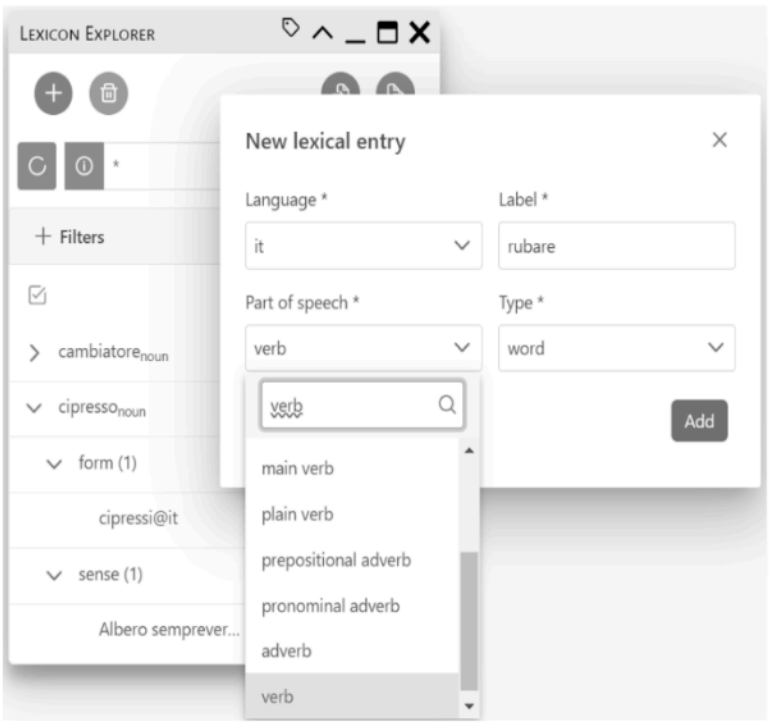

Figure 2: "Lexicon Explorer" and the entering of a Lexical Entry

As an example, we insert *rubare* (to steal); in this case, we declare that it is an Italian lexical entry of type *Word* (and not, for example, *Affix* or *Multi-word*),[19] categorised as a verb (Figure 2). Indicating the language of the entry is particularly important because the system supports the creation of multilingual lexicons.

The different lexical entries are displayed using a filterable tree structure, in which each entry constitutes the root and its subnodes are the list of forms and senses associated with it. A processing status ("working", "completed", and "reviewed") is displayed for each entry, which is useful for its management within a multi-user platform with role division.

For each lexical entry, it is possible to access the relevant instance of the "Lexicon Editor" in which we find its subtree on the left and a tab system on the right, the content of which depends on the element selected in the subtree. The subtree has two functions: on the one hand, it allows navigation between the lexical entry, its forms and senses; on the other hand, it allows the creation of new forms and senses related to the lexical entry itself. All element types displayed in the detail panel have a "Metadata" tab containing context information such as date and author of creation and any notes.

[19] The morphological and semantic description of Lexical Entries of the "Affix" type is delegated to the user and, being similar to what is expected for "Word" entries, does not require special treatment. The management of entries of "Multi-word" type, however, will be made possible once the OntoLex-Lemon "Decomposition module" will be integrated (cf. Section 4.1.1.).

For a lexical entry, "Lexical Entry" and "Relations" tabs are available. The first of these allows manipulation of a basic set of information (such as processing status and the POS), but also more complex information such as "See also" references to other lexical entries. In this case it is possible to create links not only between internal lexical entries, but also to insert a link to external lexicons through the entry of a valid identifier. The "Relations" tab, on the other hand, allows you to indicate different categories of relationships (e.g., homography), direct or indirect, with other entries in the lexicon.

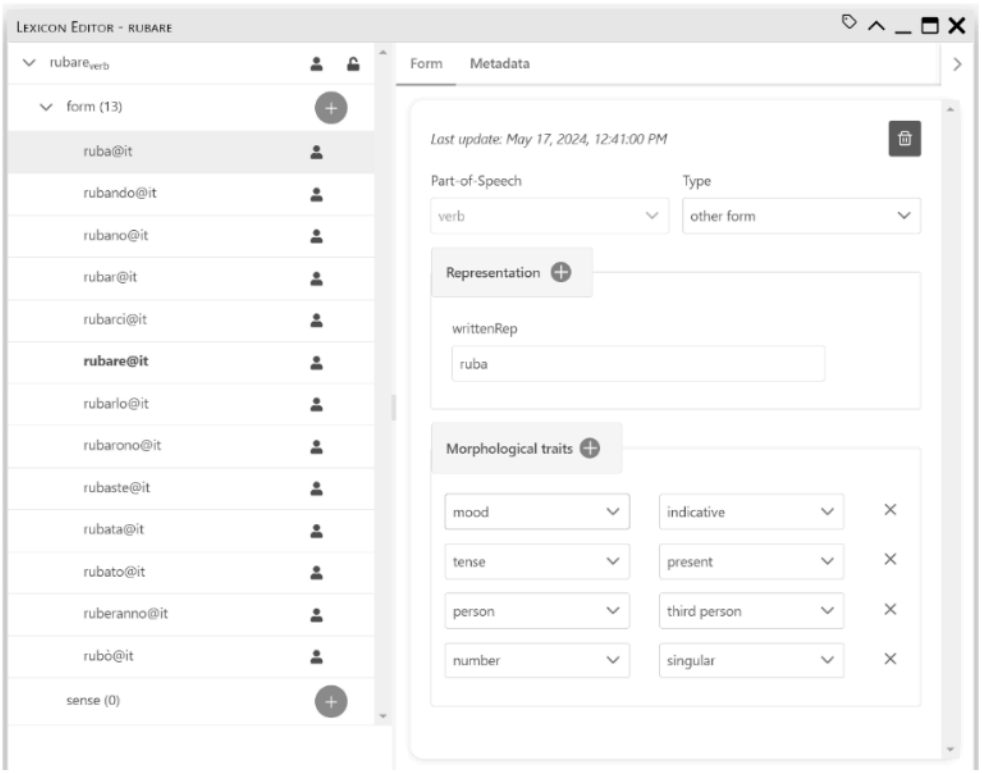

Figure 3: "Lexicon editor" for Form editing

The "Form" tab (Figure 3) allows the user to detail not only the relative written representation, but also to describe its set of morphological features. The editing of Lexical Senses is possible through two tabs: "Sense", which collects the different elements of defining a sense (definition, description, examples, etc) and "Semantic Relations", which instead allows for defining semantic relations of the sense under consideration with senses of other lexical entries. Regarding the relations categorised here as indirect, it is also possible to specify additional information, such as a confidence, comment or note.

For example, if we consider the meaning "*impossessarsi, in modo illecito e in particolare violento, di quanto appartiene ad altri*" ("to take possession, illicitly and particularly violently, of what belongs to others") for the Lexical Entry *rubare*, we can specify a synonymy relation between this and the meaning "*impossessarsi, in modo illecito e di nascosto, di ciò che appartiene ad altri*" ("to take possession, illicitly and secretly, of what belongs to others") for *imbolare* (Figure 4). Such explicitly stated relations give rise to an onomasiological network (Figure 5).

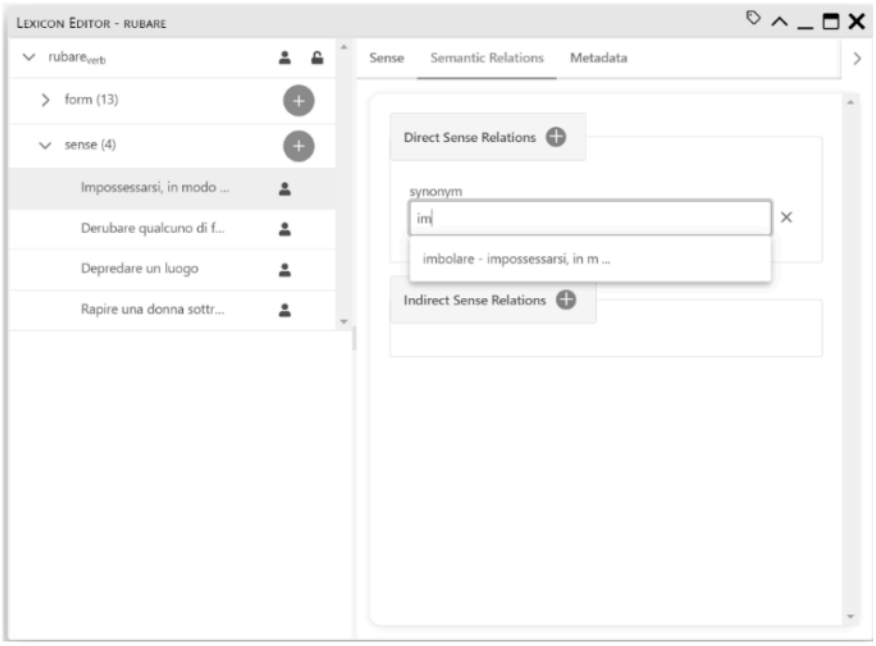

Figure 4: "Semantic Relations" tab

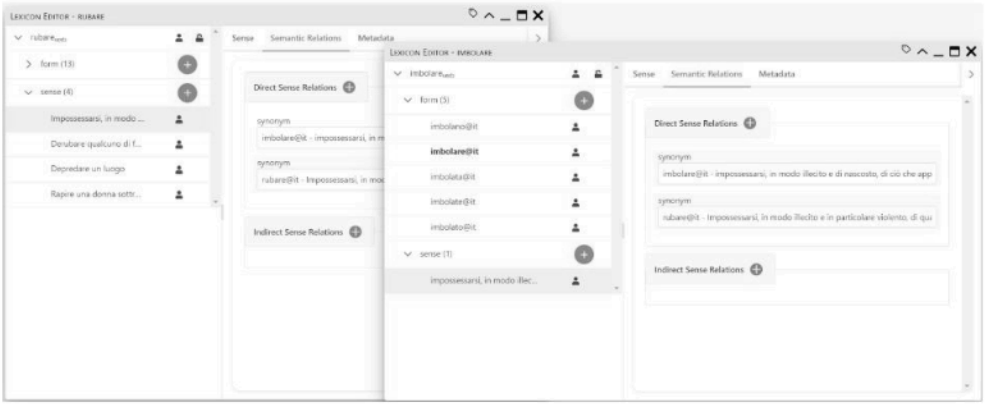

Figure 5: Example of semantic relations between two lexical senses

3.1.2. Lexicographic Resource

As with the Lexicon, the Lexicographic Resource (which in Maia is called the "Dictionary") is managed via two panels: the "Dictionary Explorer", for CRUD operations, and the "Dictionary Editor", with which to describe the Entries.

The "Dictionary Explorer" panel allows for the creation of two types of entries: the first type is a "full" lexicographic entry, which has associated lexical entries (lemmas) as its components; the second type is a "reference" entry, which lacks any components and refers to a full

lexicographic entry. Figure 6 shows the creation of the lexicographic entry *involare*, composed of the lexical entry (lemma) *imbolare*, previously created at the Lexicon level (cf. 3.1.1).[20]

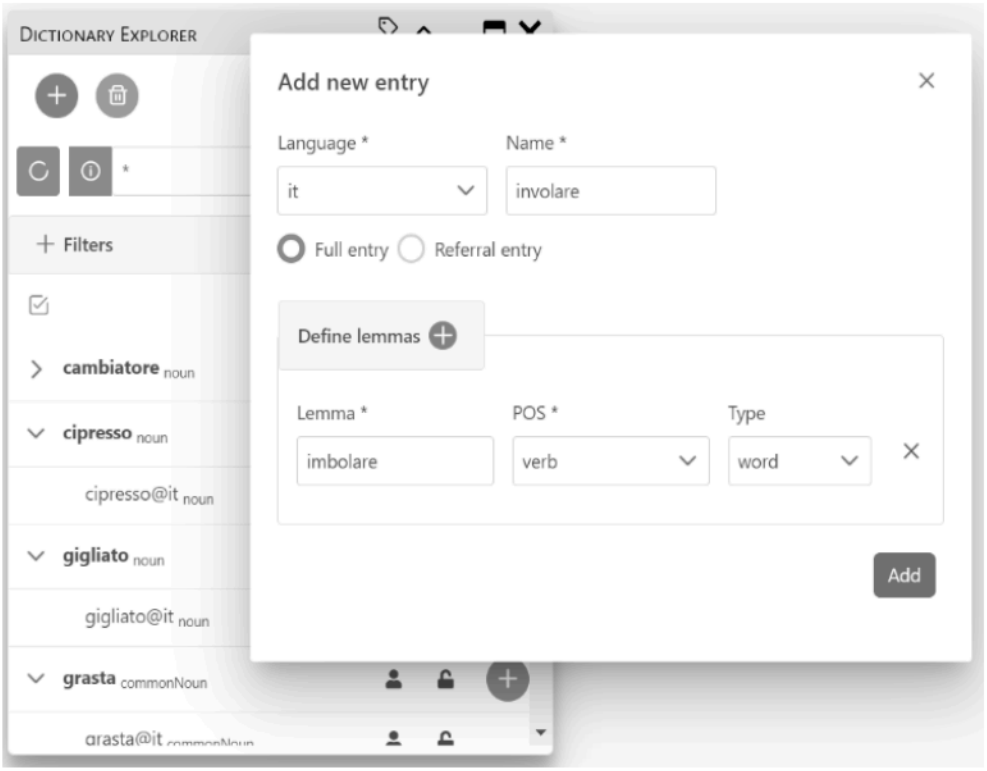

Figure 6: Creation of a lexicographic entry

As in the "Lexicon Explorer", the "Dictionary Explorer" also features a tree representation, where Entries constitute the parent nodes and Lexical Entries are the leaves. For each Entry node corresponding to a full entry, it is possible to add additional components later. Each node allows you to display the editor corresponding to the element for the detailed information compilation.

The "Dictionary Editor" aims to describe an entry, not only by compiling its basic information but also by providing functionalities such as sorting of meanings (Lexical Sense) or displaying a print preview.[21]

[20] In this case, it can be noted that the label of the lexicographic entry, involare, is formally distinct from that of the lemma that composes it, imbolare: while the contemporary Italian standard is adopted for the entry, the lemma respects the forms with "imb-" attested in Boccaccio's Decameron (reference corpus) and displayed alphabetically in the "Lexicon Editor" in Figure 5.

[21] This panel is, at the time of writing, under development.

3.2. Search and KWIC Visualization

Maia features a "Search" panel with KWIC (Keyword In Context) display of results. The functionality is available in a prototype version, which will be enhanced in subsequent updates.

The "Search" panel allows for searching by form within the corpus present in Maia (cf. 3.3), or on a subset of it, with the possibility to set the width of the displayed left and right contexts. For instance, we can search for all occurrences of *imbolare* in the text of the Decameron with contexts of 5 tokens to the left and right of the keyword (Figure 7).

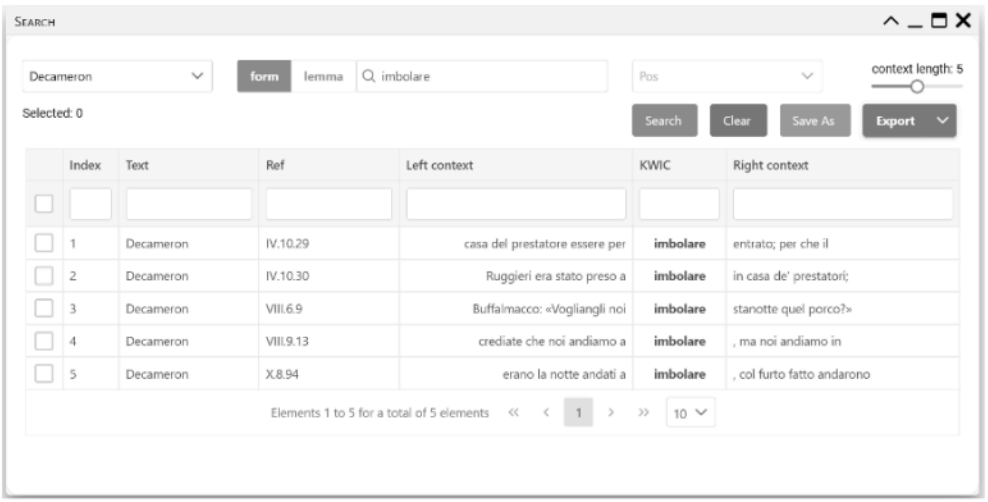

Figure 7: "Search" of form "imbolare" in the Decameron

The obtained occurrences can be further filtered using the fields in the table header. Additionally, from each occurrence, it is possible to navigate to the corresponding point in the text (cf. next section).

3.3 Text Annotation and Lexical Linking

The "Corpus Explorer" (exploratory) and "Text" (detail) panels are dedicated to text corpus management and text annotation, respectively.

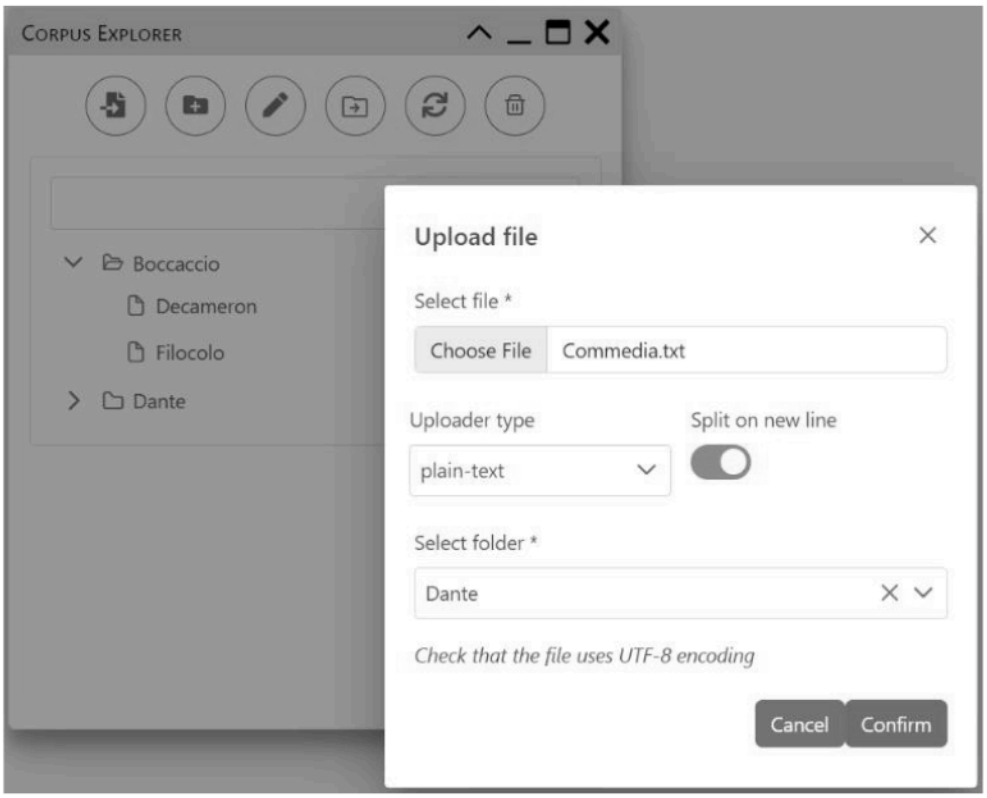

Figure 8: "Corpus Explorer" and upload of a text

The "Corpus Explorer" displays text resources in the manner of a file system, as a collection of folders and files that may be nested. The panel provides CRUD operations, allowing the user to expand the project corpus with new documents. In this initial phase, the loading of text files (.txt) encoded in UTF-8 is supported (Figure 8), possibly marked with logical divisions (for example, the text of the *Decameron* is typically marked according to the "Giornata", "Novella" and "Paragraph" structure). These divisions are used to facilitate navigation within the document (Section 4.2). During uploading, it is also possible to specify whether the line division should be based on line breaks.

A text uploaded to Maia can be annotated at various levels of granularity based on three key elements: **Layer**, which represents the annotation level and collects information describing the same aspect; **Feature**, a property associated with a layer describing its characteristics; **Tagset**, a set of predefined values that can be associated with a feature. The definition of these three types of elements is left in the hands of the users, through the two dedicated panels "Layer" and "Tagset".

For each layer, one or more features can be defined falling into the following categories: String, Tagset, URI,[22] Lexical Entry, Form, and Sense. If String-type features allows annotation with any textual content, selecting the URI type restricts the range of valid values to URIs, typically used to refer to some external resource. Similarly, when selecting the Tagset type, it needs

[22] https://en.wikipedia.org/wiki/Uniform_Resource_Identifier

specifying which one of the previously defined Tagsets is to be associated with the feature: its values will constitute the set of valid values for that attribute. Of particular importance are features of Lexical Entry, Form, and Sense types, as they realise the first level of lexical linking between the text annotation component and the construction of a lexical resource (cf. the previous section). This type of feature allows for linking a portion of text to an element from Maia's Lexicon.

If we consider the case of the *Decameron*, we may want to define two annotation layers which we'll call *Semantica* (Semantics) and *Sintassi* (Syntax). The first of these layers aims to link portions of the text to specific meanings defined in the lexicon and will be characterised by a feature of Sense type.

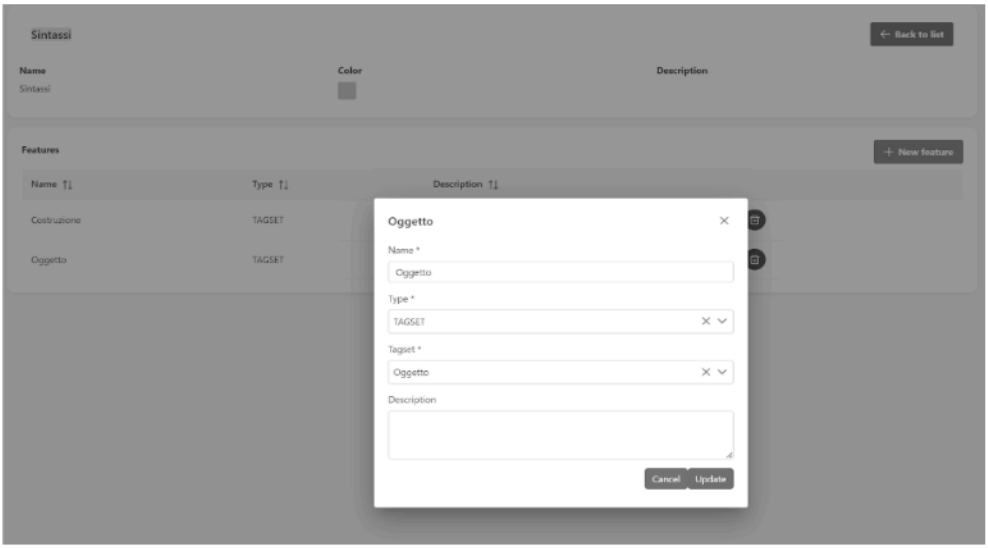

Figure 9: An example of layer: "Sintassi", with a focus on feature "Oggetto"

For layer *Sintassi* (Figure 9), for example, two features of tagset type can be used to define: i) *Costruzione* (Construction) (with two values "attiva"/"passiva", based on the active or passive form of the involved verb) and ii) *Oggetto* (Object) (with two values "personale" / "impersonale", based on the presence of a direct object consisting of a person or another entity, whether animate or inanimate: cf. Figure 10).[23]

[23] This definition of a syntactic layer is merely illustrative: each user can define their own layer through the features they deem most appropriate.

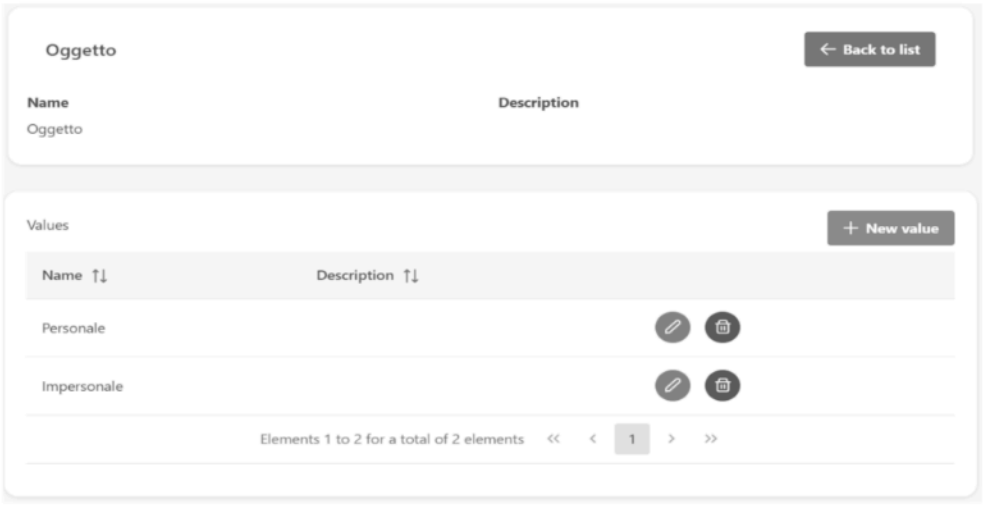

Figure 10: An example of tagset: "Oggetto", having two values.

Once the layers (with any features and tagsets) are defined, text annotation can be performed in the workspace using the "Text" panel where a text is displayed in an "infinite scroll" format,[24] divided according to the parameters defined during the upload step. The text, if structured, can be navigated with the support of an index (Figure 11, left side). The panel allows for selecting which layer to annotate with and which layers to display simultaneously (Figure 11, top from left to right). Assigning values to a single annotation is done through a dynamic form, where each editable field represents a single feature of the layer, and the type of field used depends on the type of feature (for example, a dropdown menu for tagset features). The level of granularity for text selection and annotation can go down to a single character.

[24] Infinite scroll refers to a mechanism that allows for the continuous loading of information as you scroll down the page. In the case of Maia, it is used, for example, for continuous reading of text. Initially, a first portion is loaded, and as you scroll down the page, subsequent sections are retrieved from the back-end.

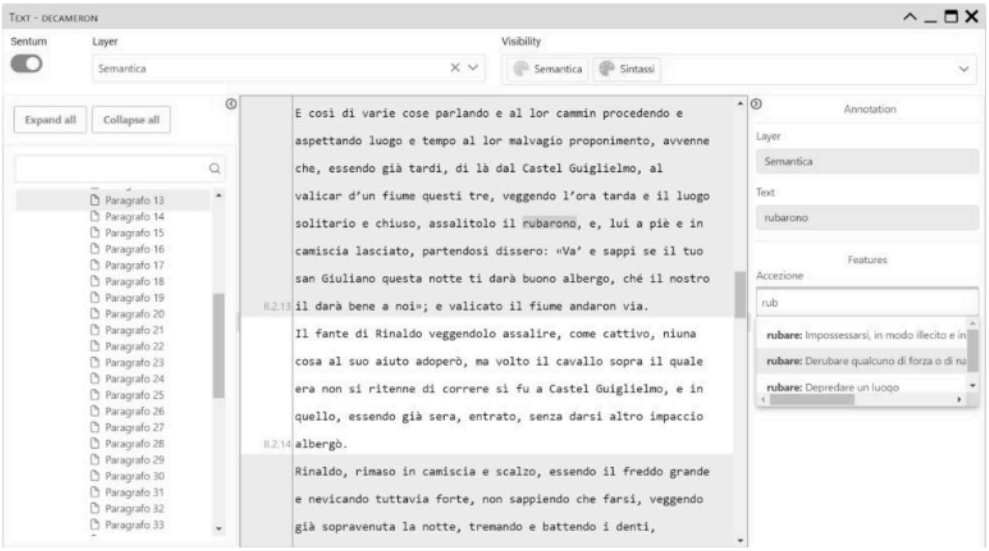

Figure 11: "Text" panel: an example of semantic annotation of the Decameron

Opening the text of the *Decameron*, we can navigate to *Giornata II*, *Novella 2*, *Paragraph 13* to annotate the occurrence of the form *rubarono* in the following context: "[...] veggendo l'ora tarda e il luogo solitario e chiuso, assalitolo il rubarono, e, lui a piè e in camiscia lasciato, partendosi dissero [...]". By choosing the layer *Semantica*, it is possible to select the portion of interest in the text: here, the string *rubarono*. In the annotation form, there is a single feature that corresponds to the feature *Accezione*: based on the user's input, this field suggests possible values to associate. It is therefore possible to link the selected text with one of the senses previously defined in the lexicon.

The text can be enriched with additional information by adding a second type of annotation, pertaining to layer *Sintassi*, even overlapping the first. The aforementioned occurrence of *rubarono* (*Decameron* II.2.13) can therefore be marked as a *Costruzione Attiva* with *Oggetto Personale*,[25] while the occurrence of *involare* in *Decameron* VIII.6.9 can be annotated as a *Costruzione Attiva* with *Oggetto Impersonale*. As it can be seen in Figure 12, the two overlapping annotations are displayed on top of each other, distinguished by the highlighting colour assigned to the respective layer.[26]

[25] This syntactic configuration of rubare versus involare reflects the norm for ancient Italian, for which cf. ([9], 90-91).

[26] A unique colour is associated to each layer during creation.

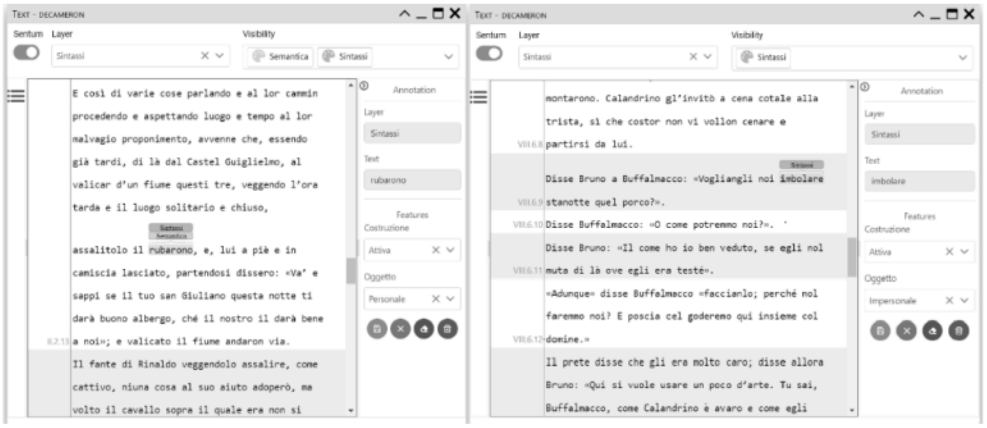

Figure 12: "Text" panel for the Decameron: syntactic annotation of "rubare" vs. "involare"

4. Software architecture and data models

Maia is based on a set of software components structured according to a modular architecture (Figure 13). As is commonly done in the development of state-of-the-art web applications, the components that make up the system can be grouped into a frontend (FE) and a backend (BE). The FE interface component runs in user's browser and is developed in Javascript, HTML, and CSS. The BE component, on the other hand, is written in Java, a more robust and typed programming language, and it interacts with various databases to provide the services required for the FE part. In order to make the architecture as modular as possible, the BE part has been further divided into modules responsible for managing the corpus and lexicon. The individual modules of the architecture are listed below:

Maia-FE: this module pertains to the user interface (UI) of Maia, whose functions have been explained in the previous section;

Maia-BE: this module manages and routes requests coming from the user interface; in particular, this module handles interactions with the other modules populating the BE, responsible for managing access, text, and the lexicon;

LexO-server: this module is responsible for managing the computational lexicon and the dictionary (for details, refer to Section 4.1);

TextO: this module manages the textual corpus and the annotations (Section 4.2).

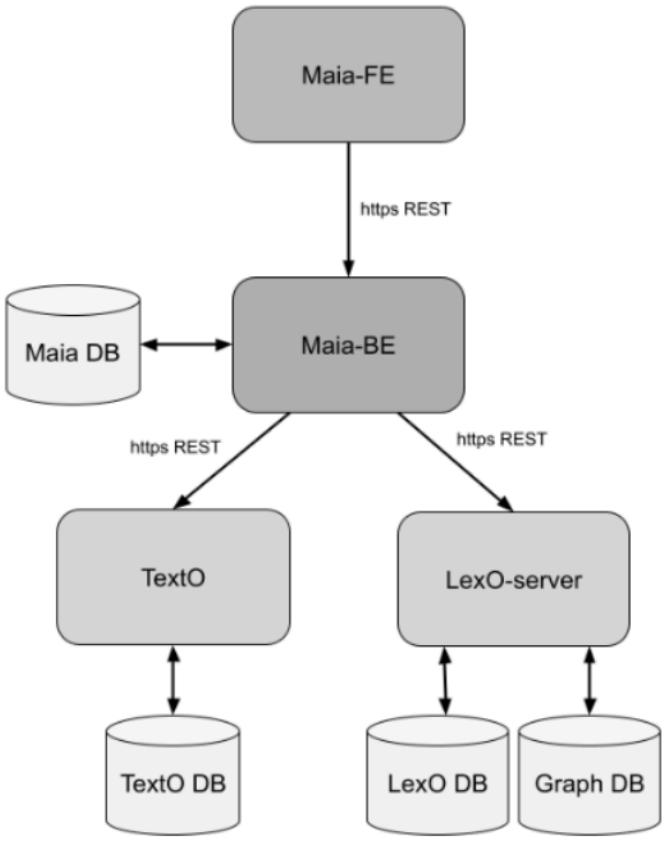

Figure 13: Maia software architecture

4.1. Lexicon management

The Semantic Web provides an opportunity to enhance the process of linguistic resource production to more effectively meet the needs of end users. In particular, it is currently possible to create lexicons that adhere to the FAIR principles (Findability, Accessibility, Interoperability, and Reuse),[27] enriched with lexical-semantic and conceptual information ([8]). In line with this perspective, Maia adopts a set of computational services called LexO-server ([1]).[28] These services allow the manipulation of data at both the lexical and conceptual levels. The morphological and semantic characteristics of the words that make up the linguistic data are formally described using the aforementioned OntoLex-Lemon model. Lexical entries can

[27] https://www.go-fair.org/fair-principles/

[28] LexO-server is a service registered within the CLARIN-IT infrastructure (http://hdl.handle.net/20.500.11752/ILC-1004)

then be organised as dictionary entries according to the Lexicographic module (lexicog) already mentioned in Section 3.1 and described in the next section.

Similarly, at the conceptual level, concepts and properties are structured and formally represented using the Simple Knowledge Organization System (SKOS) framework. Currently, LexO-server natively offers services for managing SKOS ontologies but also allows lexical entries to reference existing external OWL ontologies.[29] However, the current version of Maia only implements lexical services. In the following subsections, we will briefly describe the adopted data model and the available services for Maia.

4.1.1. OntoLex-Lemon

OntoLex-Lemon[30] is a formal representation model for lexical resources such as dictionaries and computational lexicons. It is written in the ontology modelling language OWL and provides a detailed description of word-related information, including meanings, definitions, relationships, morphological and syntactic features. The model is based on W3C and ISO standards, including LMF (Lexical Markup Framework) ([10]), LexInfo ([7])- which is aligned with ISOCat ([13]) - and LIR (Linguistic Information Repository) ([9]). The architecture of OntoLex-Lemon is divided into five modules, each handling specific aspects in the modelling of lexical information: the Core module (for describing the structure of lexical entries, already introduced in Section 3), the Decomposition module (for describing multiword compounds), the Variation and Translation module (for describing lexical-semantic relations), the Syntax and Semantics module (for describing syntactic aspects), and the Metadata module (for representing quantitative and qualitative data about the resource). For details on the various modules, refer to the W3C Community Group final report (cf. footnote 30).

Recently, the same Community Group also developed a module for dictionary management called "Lexicog".[31] The main objective of this module is to complement the core module of OntoLex-Lemon by offering a different organisation of data, specifically that of a dictionary. The model aims to provide common characteristics of dictionaries by modelling them in a way that is independent of specific linguistic perspectives. Naturally, some specific elements of dictionaries will still require ad hoc constructs. However, the set of common entities defined in the module helps to overcome the limitations that have been observed in experiences of instantiating OntoLex-Lemon with lexicographic data. In particular, through this module it is possible to: i) organise lexical entries into dictionary entries, thus grouping different grammatical categories; ii) sort the lexical senses belonging to entries; iii) specify sub-senses.

4.1.2. LexO-server

LexO-server[32] is a free and open-source backend for managing lexical and ontological resources. It is built upon the semantic repository called GraphDB.[33] LexO-server is implemented as a set of REST services based on the HTTP protocol and exchanges data in JSON format. It originated from the experience gained in the development of LexO-lite ([2]), a full-stack tool for managing OntoLex-Lemon resources. LexO-server enables the management

[29] https://it.wikipedia.org/wiki/Web_Ontology_Language

[30] https://www.w3.org/2016/05/ontolex/

[31] https://www.w3.org/2019/09/lexicog/

[32] https://github.com/andreabellandi/LexO-backend

[33] https://www.ontotext.com/

of both lexical and conceptual levels and allows connecting these two levels through links between words and concepts.

LexO-server services provide various functionalities, including: i) editing of lexical, terminological, and lexicographic resources; ii) navigating a resource as a graph (calculating the shortest path between senses with respect to a specific relationship, calculating the distance between senses to distinguish whether the relationships are inferred or explicit, etc.); iii) supporting linguistic-semantic access to texts, using the resource to trace from forms to lemmas and from lemmas to the concepts they denote.

Maia utilises the set of LexO-server services for entity editing features in OntoLex-Lemon and advanced searches to enable summary data visualisations, such as lists or trees, as well as detailed tables or input forms for editing purposes.

4.2. Corpus and annotations management

Both the corpus and the annotations on the relative texts are managed by the same BE module called TextO,[34] implemented using the Spring Boot[35] development framework and made available as an open-source software.

Firstly, TextO allows the uploading of textual documents into a MySQL[36] relational database and their organisation into folders and subfolders, which can be created, moved, and deleted, similarly to how files are managed by an operating system.

Currently, as anticipated in Section 3.3, the import services allows to upload UTF-8 encoded "plain-text" documents or "marked-plain-text" documents (see below), in this latter case with the option of having them also segmented on carriage returns. Documents can be marked to describe their structure: during the import phase, TextO parses the text and interprets the markings to divide it into logical, possibly nested, parts, as previously shown in the example of the text of the *Decameron* in Section 3.3. The mark-down syntax, described in more detail on the TextO Git page, specifies the depth of the structural level using a number of "#" characters (to allow for sub-level organisation), the type (e.g., "chapter", "paragraph", etc.), a title (which will be visible on the left side of the "Text" panel), and a unique index that identifies the section (used internally by Maia for navigation between sections). Here is an example of structural organisation of the *Decameron*:

#(type="Giornata", title="Giornata I", index="I")

I

##(type="Novella", title="Novella 1", index="I.1")

I.1

###(type="Paragrafo", title="Paragrafo 1", index="I.1.1")

Ser Cepparello con una falsa confessione inganna un santo frate e muorsi [...]

###(type="Paragrafo", title="Paragrafo 2", index="I.1.2")

[34] https://github.com/davide-albanesi-ilc/TextO

[35] https://spring.io/projects/spring-boot

[36] https://www.mysql.com

Convenevole cosa è, carissime donne, che ciascheduna cosa la quale l'uomo fa […]

Import services that will enable the uploading of texts with different formats (e.g., html, pdf, docx, etc.), structural mark-ups (such as with TEI-XML), and that present existing annotations (e.g. CoNLL) are under development at the time of writing of this article.

From a user perspective, TextO can already manage multiple accounts, each of which is assigned its own storage space for its data. The storage space can also be shared among multiple users, allowing the sharing of individual texts or entire folders with collaborators. The sharing is selective: the system enables sharing a text with one's own annotations in read-only mode or making the annotations modifiable.

Regarding search functionalities, TextO exposes services to query the corpus by keyword, on the basis of the available annotations and considering a left and right context of search of arbitrary lenght. At the time of writing, however, Maia's front-end only allows to search by keyword and view the results in KWIC format (cf. Section 3.2).

As evident from the interface presentation, TextO manages, in addition to texts, the annotations and the related entities associated with them, such as "Layers", "Features", and "Tagsets". All these data are stored in the already cited MySQL database (TextO DB of Figure 13). Annotations, in particular, are represented by storing the position of the first and last character of the textual fragment being annotated, thus allowing an easily management of partially overlapping annotations.

5. Conclusions and perspectives

This article introduced Maia, a collaborative, open-source web tool for text annotation, lexicon construction, and lexical linking primarily designed for use by digital humanists. As a matter of fact, Maia was conceived, designed, and developed by capitalising on the authors' years of experience gained through various DH projects, both national and international, focused on building lexical and terminological resources. These include the DiTMAO project (Dictionary of Old Occitan medico-botanical terminology) ([3]), ([26]),[37] the Totus Mundus project ("Todo el mundo es nuestra casa", The World is our Home: A Virtual Journey Around the World Atlas by Matteo Ricci) ([19]),[38] the "Per una edizione digitale dei manoscritti di Ferdinand de Saussure" project ([11]),[39] and, most recently, the "Representing Religious Diversity in Europe: Past and Present & Features" project ([22]).

A common factor in all of these projects has been the expressed need of scholars to work within the same environment, combining a reference corpus for resource construction and formal connections between lexical elements and textual elements. It is from this need that it was deemed appropriate to develop a collaborative working environment that encompasses both text and lexicon, allowing for the annotation of the text with arbitrary information (organised in layers) and constructing the lexicon based on established lexical models. Through linking between text annotations and lexicon elements, it becomes possible to establish a

[37] https://www.uni-goettingen.de/en/487498.html

[38] http://wafi.iit.cnr.it/annotarium/#/totusmundus_vfs%7Ctotusmundus_root

[39] https://www.ilc.cnr.it/en/progetti/saussure-2/

formal connection between lexical entries, forms or senses, and their respective occurrences in a reference corpus, which can be structured into folders and subfolders.

The advantages of having an integrated environment like the one provided by Maia are manifold. Firstly, as previously mentioned, within a task of constructing (or revising) a lexical or terminological resource, it is possible to consult the documents used as textual sources within the same environment. The ability to search through texts, annotate specific parts, and link these annotations to elements in the lexicon can, for instance, allow the definition of a polysemous term in its various meanings and connect the various senses created to their respective occurrences in the text. This way, it will be possible to have a series of contexts for each lexical entry, form, or sense defined in the lexicon, both for internal reference within the tool and for potential exports as examples of occurrences in the form of electronic dictionaries.

Furthermore, the collaborative nature of Maia allows for the creation of shared work projects (whether they involve text annotation, lexicon construction, or a combination of both) involving multiple scholars, each of whom can focus on specific portions of the dataset, enabling parallel work.

The development prospects for Maia, presented and released in its initial version in this article, as described in Section 3, are multifaceted. Firstly, the search functionalities will be extended, both for the corpus and the lexicon, and a dedicated section for linguistic-based search supported by the lexical resource will be developed, as experimented in ([12]).

Secondly, there are plans to include the ability to create and edit ontologies within the environment. This work, already in progress, will allow for direct connections of text annotations (e.g., "named entity" types) to concepts within the ontology and linking senses from the lexical resource to these concepts. This will introduce a conceptual layer into the overall resource, distinct from the linguistic-semantic layer. In the first instance, Maia will allow importing and visualising OWL ontologies while the editing part will be developed later.

Maia will also incorporate data import and export features for the textual corpus and lexical resources. Initially, as anticipated, it will be possible to import pre-annotated texts in a format similar to CoNLL (which can then be manually reviewed in the system) and lexical resources in RDF format compliant with the OntoLex-Lemon model[40]. The data will, therefore, be exportable in the same formats. Subsequently, additional import and export formats will be included to accommodate the needs of users from various communities. Regarding the export of the annotated corpus, we plan to adopt two distinct models. To represent annotations we chose to use the Web Annotation Data Model[41], while to formalise textual data we'll adopt POWLA[42] (Portable Linguistic Annotation with OWL), an OWL vocabulary for linguistic annotations based on the ISO TC37/SC4 Linguistic Annotation Framework standard.

Several upgrades are already in progress involving specifically the annotation component. To facilitate the annotation work in a collaborative context, TextO already includes versioning

[40] However, in the current version of Maia, as illustrated here, it is already possible to import an OntoLex-Lemon lexicon by uploading it directly into the GraphDB database, although it is not yet possible to do so through the user interface.

[41] https://www.w3.org/TR/annotation-model/

[42] https://github.com/acoli-repo/powla

services. Specifically, whenever a text is modified (by adding, editing, or removing an annotation), the system saves the previous version before making the change. This way, all versions of a text are preserved and can be restored as needed. Finally, a feature is being developed that will allow for the creation of links between annotations, similar to what the INCEpTION tool already allows. Through this feature, it will be possible to define, for example, anaphoric or syntactic relationships.

Funding

This work was carried out in the context of project PRIN 2017 "Representing Religious Diversity in Europe: Past and Present Features", project "Rut - modelli, risorse, metodologie e strumenti per la rappresentazione di risorse terminologiche e ontologiche", project VocaBO - "Vocabolario di Boccaccio Online", and the Babylonian Talmud Translation project within the scientific collaboration between S.c.a r.l. PTTB and CNR-ILC (09/11/2017).

References

[1] Bellandi, Andrea. 2023. "Building Linked Lexicography Applications with LexO-Server." *Digital Scholarship in the Humanities* 38 (3): 937–52. https://doi.org/10.1093/llc/fqac095.

[2] Bellandi, Andrea. 2021. "LexO: An Open-Source System for Managing OntoLex-Lemon Resources." *Language Resources and Evaluation* 55 (4): 1093–1126. https://doi.org/10.1007/s10579-021-09546-4.

[3] Bellandi, Andrea, Emiliano Giovannetti, and Anja Weingart. 2018. "Multilingual and Multiword Phenomena in a Lemon Old Occitan Medico-Botanical Lexicon." *Information* 9 (3): 52. https://doi.org/10.3390/info9030052.

[4] Bergamaschi, Sonia, Laura Po, Serena Sorrentino, and Alberto Corni. 2010. "Dealing with Uncertainty in Lexical Annotation." *Revista de Informática Teórica e Aplicada* 16 (2): 93–96. https://doi.org/10.22456/2175-2745.12580.

[5] Bizer, Christian, Tom Heath, and Tim Berners-Lee. 2009. "Linked Data: The Story so Far." *International Journal on Semantic Web and Information Systems* 5 (July):1–22. https://doi.org/10.4018/jswis.2009081901.

[6] Chiarcos, Christian, and Maxim Ionov. 2019. "Ligt: An LLOD-Native Vocabulary for Representing Interlinear Glossed Text as RDF." Application/pdf. *OASIcs, Volume 70, LDK 2019* 70:3:1-3:15. https://doi.org/10.4230/OASICS.LDK.2019.3.

[7] Cimiano, P., P. Buitelaar, J. McCrae, and M. Sintek. 2011. "LexInfo: A Declarative Model for the Lexicon-Ontology Interface." *Journal of Web Semantics* 9 (1): 29–51. https://doi.org/10.1016/j.websem.2010.11.001.

[8] Costa, Rute, Ana Salgado, and Bruno Almeida. 2021. "SKOS as a Key Element for Linking Lexicography to Digital Humanities." In *Information and Knowledge Organisation*

in Digital Humanities, by Koraljka Golub and Ying-Hsang Liu, 1st ed., 178–204. London: Routledge. https://doi.org/10.4324/9781003131816-9.

[9] Espinoza, Mauricio, Asunción Gómez-Pérez, and Elena Montiel-Ponsoda. 2009. "Multilingual and Localization Support for Ontologies." In *The Semantic Web: Research and Applications*, edited by Lora Aroyo, Paolo Traverso, Fabio Ciravegna, Philipp Cimiano, Tom Heath, Eero Hyvönen, Riichiro Mizoguchi, Eyal Oren, Marta Sabou, and Elena Simperl, 5554:821–25. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-02121-3_63.

[10] Francopoulo, Gil, Monte George, Nicoletta Calzolari, Monica Monachini, Nuria Bel, Mandy Pet, and Claudia Soria. 2006. "Lexical Markup Framework (LMF)." In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, edited by Nicoletta Calzolari, Khalid Choukri, Aldo Gangemi, Bente Maegaard, Joseph Mariani, Jan Odijk, and Daniel Tapias. Genoa, Italy: European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2006/pdf/577_pdf.pdf.

[11] Gambarara, Daniele, and Maria Pia Marchese, eds. 2013. *Guida per Un'edizione Digitale Dei Manoscritti Di Ferdinand de Saussure: Progetto Di Ricerca PRIN 2008*. Studi e Ricerche 117. Alessandria: Edizioni dell'Orso.

[12] Giovannetti, Emiliano, Davide Albanesi, Andrea Bellandi, Simone Marchi, Mafalda Papini, and Flavia Sciolette. 2022. "The Role of a Computational Lexicon for Query Expansion in Full-Text Search." In *Proceedings of the Eighth Italian Conference on Computational Linguistics CliC-It 2021*, edited by Elisabetta Fersini, Marco Passarotti, and Viviana Patti, 162–68. Accademia University Press. https://doi.org/10.4000/books.aaccademia.10638.

[13] Kemps-Snijders, Marc, Menzo Windhouwer, Peter Wittenburg, and Sue Ellen Wright. 2008. "ISOcat: Corralling Data Categories in the Wild." In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, edited by Nicoletta Calzolari, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, and Daniel Tapias. Marrakech, Morocco: European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/pdf/222_paper.pdf.

[14] Klie, Jan-Christoph, Michael Bugert, Beto Boullosa, Richard Eckart de Castilho, and Iryna Gurevych. 2018. "The INCEpTION Platform: Machine-Assisted and Knowledge-Oriented Interactive Annotation." In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, 5–9. Santa Fe, New Mexico: Association for Computational Linguistics. https://www.aclweb.org/anthology/C18-2002.

[15] Liebeskind, Chaya, Ido Dagan, and Jonathan Schler. 2018. "Automatic Thesaurus Construction for Modern Hebrew." In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, edited by Nicoletta Calzolari, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Koiti Hasida, Hitoshi

Isahara, et al. Miyazaki, Japan: European Language Resources Association (ELRA). https://aclanthology.org/L18-1229.

[16] Litkowski, K.C. 2006. "Computational Lexicons and Dictionaries." In *Encyclopedia of Language & Linguistics*, 753–61. Elsevier. https://doi.org/10.1016/B0-08-044854-2/00945-7.

[17] Měchura, Michal. 2017. "Introducing Lexonomy: An Open-Source Dictionary Writing and Publishing System." In *Electronic Lexicography in the 21st Century: Proceedings of eLex 2017 Conference*, edited by V. Baisa I. Kosem C. Tiberius, M. Jakubíček, J. Kallas, S. Krek, 662–79. Leiden: Lexical Computing. https://elex.link/elex2017/.

[18] Mengaldo, Pier Vincenzo. 1961. "'Involare e rubare' in italiano antico / Pier Vincenzo Mengaldo." In *"Involare e rubare" in italiano antico*. Firenze: Sansoni.

[19] Piccini, Silvia, Elisabetta Corsi, and Emiliano Giovannetti. 2017. "Une ressource termino-ontologique bilingue chinois-italien: le cas de la traduction de la Mappemonde de Matteo Ricci par Pasquale D'Elia." In *Actes de la conférence TOTh 2017*, 33–49. Terminologica. Chambéry: Université Savoie Mont Blanc. http://ontologia.fr/TOTh/Conference/TOTh2017/TOTh_2017.pdf.

[20] Romary, Laurent. 2015. "TEI and LMF Crosswalks." *Journal for Language Technology and Computational Linguistics* 30 (1): 47–70. https://doi.org/10.21248/jlcl.30.2015.195.

[21] Romary, Laurent, and Toma Tasovac. 2018. "TEI Lex-0: A Target Format for TEI-Encoded Dictionaries and Lexical Resources." In *TEI Conference and Members' Meeting*. Tokyo, Japan. https://inria.hal.science/hal-02265312.

[22] Saponaro, D., E. Giovannetti, and F. Sciolette. 2022. "From Religious Sources to Computational Resources: Approach and Case Study on Hebrew Terms and Concepts." *Materia Giudaica Print* XXVII(2022):21.

[23] Singh, Siddharth, Ritesh Kumar, Shyam Ratan, and Sonal Sinha. 2022. "Towards a Unified Tool for the Management of Data and Technologies in Field Linguistics and Computational Linguistics - LiFE." In *Proceedings of the Workshop on Resources and Technologies for Indigenous, Endangered and Lesser-Resourced Languages in Eurasia within the 13th Language Resources and Evaluation Conference*, edited by Atul Kr. Ojha, Sina Ahmadi, Chao-Hong Liu, and John P. McCrae, 90–94. Marseille, France: European Language Resources Association. https://aclanthology.org/2022.eurali-1.16.

[24] Stellato, Armando, Manuel Fiorelli, Andrea Turbati, Tiziano Lorenzetti, Willem Van Gemert, Denis Dechandon, Christine Laaboudi-Spoiden, et al. 2020. "VocBench 3: A Collaborative Semantic Web Editor for Ontologies, Thesauri and Lexicons." Edited by Aidan Hogan. *Semantic Web* 11 (5): 855–81. https://doi.org/10.3233/SW-200370.

[25] Uren, Victoria, Philipp Cimiano, Jose Iria, Siegfried Handschuh, Maria Vargas-Vera, Enrico Motta, and Fabio Ciravegna. 2006. "Semantic Annotation for Knowledge Management: Requirements and a Survey of the State of the Art." *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3199324.

[26] Weingart, Anja, and Emiliano Giovannetti. 2016. "Extending the Lemon Model for a Dictionary of Old Occitan Medico-Botanical Terminology." In *The Semantic Web*,

edited by Harald Sack, Giuseppe Rizzo, Nadine Steinmetz, Dunja Mladenić, Sören Auer, and Christoph Lange, 9989:408–21. Lecture Notes in Computer Science. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-47602-5_53.

[27] Wilby, David, Twin Karmakharm, Ian Roberts, Xingyi Song, and Kalina Bontcheva. 2023. "GATE Teamware 2: An Open-Source Tool for Collaborative Document Classification Annotation." In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, edited by Danilo Croce and Luca Soldaini, 145–51. Dubrovnik, Croatia: Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.eacl-demo.17.

[28] Zenzaro, S., A. M. Del Grosso, F. Boschetti, and G. Ranocchia. 2023. "Ease the Collaboration Making Scholarly Editions: The GreekSchools Case Study." In *Atti Del XII Convegno Annuale AIUCD*, edited by E. Carbé, Alessia Lo Piccolo Gabrieleand Valenti, and F. Stella, 230–32. Siena: Alma Mater Studiorum-Università di Bologna (Bologna, ITA). https://doi.org/10.6092/unibo/amsacta/7721.

[29] Zenzaro, S., A. M. Del Grosso, F. Boschetti, and G. Ranocchia. 2022. "Verso La Definizione Di Criteri per Valutare Soluzioni Di Scholarly Editing Digitale: Il Caso d'uso GreekSchools." In *AIUCD 2022-Proceedings. Culture Digitali. Intersezioni: Filosofia, Arti, Media*, edited by F. Ciracì, G. Miglietta, and C. Gatto, 20–25. https://doi.org/10.6092/unibo/amsacta/6848.

[30] Zhang, Chao, Fangbo Tao, Xiusi Chen, Jiaming Shen, Meng Jiang, Brian Sadler, Michelle Vanni, and Jiawei Han. 2018. "TaxoGen: Unsupervised Topic Taxonomy Construction by Adaptive Term Embedding and Clustering." In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2701–9. London United Kingdom: ACM. https://doi.org/10.1145/3219819.3220064.