A generic hybrid method combining rules and machine learning to automate domain independent ontology population

Yohann Chasseray, Anne-Marie Barthe-Delanoë, Jérôme Volkman, Stéphane Negny, Jean-Marc Le Lann

A generic hybrid method combining rules and machine learning to automate domain independent ontology population

Yohann Chasseray [a,*], Anne-Marie Barthe-Delanoë [a], Jérome Volkman [b], Stéphane Négny [c], Jean Marc Le Lann [c]

[a] Centre Génie Industriel, IMT Mines Albi-Carmaux, Campus Jarlard, Albi, 81000, Occitanie, France
[b] Laboratoire de Chimie de Coordination, Université de Toulouse, CNRS, INPT, UPS, 4 allée Emile Monso, Toulouse, 31000, Occitanie, France
[c] Laboratoire de Génie Chimique, Université de Toulouse, CNRS, INPT, UPS, 4 allée Emile Monso, Toulouse, 31000, Occitanie, France

A B S T R A C T

Knowledge management has become a cornerstone of decision support and system engineering. Knowledge acquisition has traditionally been performed manually, and the trend now is to automate knowledge extraction from the huge amount of information contained in daily produced data. This article proposes a contribution in the artificial intelligence domain through a hybrid approach for the discovery of concept-instance couples to populate an ontology. The proposed framework combines automated domain-independent rule-based extraction for unsupervised relation extraction and semantic-oriented machine learning techniques for knowledge base enrichment. In the engineering field, another contribution resides in the generic aspect of the framework, leading to the possibility to populate ontologies and automatically build knowledge bases in various domains. The case study supporting this framework and its technical implementation show that the proposed method can be applied identically (1) to different data sources and (2) with different ontologies, regardless of the domain or subdomain they describe or the structure they have. Changing these inputs can be done without affecting the performance of the rule-based extraction, which is around 60% in terms of precision. Three different matching methods are also presented. Their ability to match new instances to their corresponding ontological class (or concept) is evaluated through a case study on biochemistry annotated textual data. The best matching method achieves an average precision score of 70% and an average recall of 74%.

1. Introduction

From industries to individuals, and from world-scale companies to small businesses, everyone produces data. Within these heterogeneous data, everyone produces knowledge. On the other side, if one considers a system, whatever it is (a chemical plant, a humanitarian crisis, an assembly line), knowledge about this system is the first necessity to understand it, interact with it and finally make decisions that impact it. Since the emergence of the Semantic Web, the formalization of ontologies, and the growing use of inference engines, many fields of research have been focused on ontology-driven decision systems. From medical science to crisis management, the literature is full of decision systems based on ontological reasoning. Building a domain-related ontology is considered a process in itself, requiring both domain experts and ontology management experts in order to organize the knowledge of a domain into concepts and properties. This process of conceptualization is definitely required for anyone who wants to build an ontology that covers the entirety of a domain. Although the decision systems mentioned above provide some valuable help and information to decision-makers, they also have a major remaining weakness: they require a populated (i.e. instantiated) ontology to be run on specific cases. However, the step of manually instantiating an ontology remains one of the largest time and resource-consuming tasks in the ontology management domain. In addition, while an ontology aims to represent a very high-level view of a domain and is suitable for any specific case that sticks to that domain, the instantiated version of the same ontology will necessarily fluctuate with the different use cases it is meant to describe. Since we have these changes, the instantiation of an ontology cannot be done once but must be done each time a single use case is considered.

1.1. Automation of the ontology population process

To facilitate the population of an ontology and to make it easily replicable, the trend is towards the automation of the instantiation

process. Harvesting several distinct data sources (Wikipedia articles, medical report database, verbatims, etc.) thus means facing many obstacles, among which in particular (i) the heterogeneity of the data, (ii) the plurality of sources and (iii) the unstructured nature of the latent information.

Despite a large number of research topics already dedicated to the task of automating ontology population, most of it remains tied to a specific domain (health, agriculture, complex event processing) and provides domain-centric solutions. Consequently, the systems proposed in these approaches are hardly transferable or replicable to other domains, or even to a different use case within the same domain.

Other strategies succeed in implementing a generic pipeline for the building of an ontology from scratch rather than populating existing ontologies. This kind of approach can be very suitable and appropriate to avoid the step of creating an ontology in domains where no ontology has been proposed yet. Nonetheless, for domains where an ontology already exists, these approaches deprive themselves of the knowledge contained in that ontology.

1.2. Proposal

Two main concerns have been identified in the field of automated ontology population, which are (i) the need for genericity in the approach in order to remain independent from the domain and (ii) the need for data source coverage due to the variability aspect of big data. The aim of this article is to present unsupervised and independent extraction techniques to extract ontology-related instances from textual data. Each of these techniques shares the following common properties:

- They are designed to be used in an unsupervised context. This means that they can be applied to a domain of expertise even if that domain is relatively poor in terms of labelled and annotated data.
- They take advantage of the knowledge contained in predefined ontologies that need to be instantiated without being bound to a specific ontology.
- They remain generic regarding the domain described by the ontology. This means that, no matter which ontology is used for instantiation, the extraction techniques will remain effective.

This article focuses on hyponymy relation extraction from textual data. However, the same methodology can be adapted and applied to differently formatted data such as structured text, databases or even images.

Besides being adaptable to distinct ontologies, the framework uses the knowledge they contain. This knowledge lies in the concepts and relations between concepts that are defined within the ontology to be instantiated. Since they are the result of a previous domain-specific knowledge acquisition process, they constitute a primary reference for the instantiation process.

The contributions conveyed by the proposed extraction framework are presented through two main blocks, which are (i) the definition of an information extraction pipeline and algorithm supported by natural language processing methods (ii) the extension of the extraction pipeline with a semantic retroactive loop used to broaden the extraction spectrum.

The remainder of this article is structured as follows. In Section 2 we present the past and ongoing related work on the subject of ontology management. In Section 3, we introduce our general iterative framework for collecting information from heterogeneous data sources and we present the different processing pipelines and methods involved. In Section 4 we apply our methodology to a chemical case study. Sections 4.4 and 4.5 respectively present the results of the application to this case study and provide the evaluation of respectively, rule-based extraction and semantic loop. In Section 5 we open the discussion about current limitations and future enhancements of the framework.

2. Related work

An ontology, as conceptualized by Gruber (1993), is a formal representation of knowledge within a specific domain, encompassing concepts, relationships, and properties. This formalization aims at facilitating knowledge sharing and interoperability among different systems and applications. As many ontologies are available in various domains, most of them remain unused because filling them manually with instances is a time-consuming task. This section lists and compares the different existing approaches for the acquisition of a populated ontology from unstructured text. The comparison, summed up in Table 1 also reveals the interest of a combined approach including rule-based extraction and machine learning techniques to fully automate the extraction of knowledge without using prior data while remaining domain independent.

2.1. Ontology learning, ontology enrichment, ontology population

The term of ontology management refers to different tasks that include the building of an ontology and the extension of an ontology or a knowledge base (Konys, 2018, 2022). The design of an ontology is a very complex task because it involves both technical domain related knowledge and ontological design skills (Nicola et al., 2009).

Some studies, such as Louge et al. (2018), Paukkeri et al. (2012), Rani et al. (2017), all cope with the issue of ontology learning and address the problem of automatic or semi-automatic building of ontologies from domain-related knowledge resources. As underlined by Khadir et al. (2021), all the proposed ontology-building methods follow a similar pattern, involving (i) concept detection in data followed by (ii) extraction of relations involving these patterns. Khadir et al. (2021) also insist on the fact that none of the current methods allow building an actionable ontology without involving an expert to redesign it.

Other studies, such as Arnold and Rahm (2014), Bosselut et al. (2019), Rajput and Gurulingappa (2013), Vicient et al. (2013), focus on the completion and enrichment of existing knowledge bases using ontology matching techniques such as string-based similarity measures or semantic-based similarity measures. All these methods have shown effectiveness and good performances in several domains for the automated construction or completion of an ontology directly from data. However, they cannot be assimilated to the task of ontology population because they either modify the structure of the ontology they enrich or build a whole ontology from scratch. Similarly, graph knowledge building which is a close research area also proposes the creation of structured knowledge. However, despite being adaptable to several domains, these approaches never refer to an existing ontology and often build the knowledge structure from scratch. In their approach (Leshcheva and Begler, 2022) propose to map the extracted knowledge graph to the targeted domain ontology after an initial formatting ontology-guided extraction step.

2.2. Information and knowledge extraction methods for ontology population

To perform the population of an ontology, the main task is to find instances of a given concept, that is, a specific occurrence of a concept. For example, one can consider that a car is an instance of the higher-level concept vehicle, or that red wine is an instance of the more generic concept wine. The relation between a generic concept and a more specific form of that concept (i.e., an instance) is a relation of hyponymy (also called *is-a* relation). When collecting instances, one of the main tasks is then to identify the occurrence of hyponymy relations.

Table 1
Comparison of approaches for the acquisition of a populated ontology from unstructured text. (OE : Ontology Enrichment, OL : Ontology Learning, OP : Ontology Population).

Reference	Method	Prior need	Data specificity	Ontology vol.	Domain dep.	Task
Alec (2023)	Rule based extraction	None	Classified ads	NA	Medium	OP
Arnold and Rahm (2014)	Ontology matching tool	Back. Know.	None	> 5 000 cpt.	Low	OL
Paukkeri et al. (2012)	Self Organizing Map (Clustering)	Not needed	Describes concepts	166 cpt.	Low	OL
Bosselut et al. (2019)	Language Model (Transformers)	Prior triples	Task specific	NA	Low	OE
De Silva and Jayaratne (2009)	K-Means, Rule based extraction	None	Wikipedia articles	NA	Medium	OL
Geng et al. (2020)	Neural Network (LSTM)	Ann. dataset	None	17 cpt., 9 rel.	High	OP
Kaushik and Chatterjee (2018)	Pattern Extraction	None	Task specific	1 cpt., 4 rel.	High	OP
Leshcheva and Begler (2022)	ontology guided extraction	None	Semi structured	3 cpt., 3 rel.	High	OP
Lomov et al. (2020)	Language Model (NN)	Train. couples	None	NA	High	OE
Louge et al. (2018)	DBSCAN (Clustering)	None	Task specific	100 cpt.	Medium	OL
Pennacchiotti and Pantel (2006)	Rule based extraction	None	None	NA	Low	OP
Rajpathak (2013)	Co-occurrence analysis	None	Repair verbatims	6 cpt., 7 rel.	High	OP
Rajput and Gurulingappa (2013)	Background ontology querying	Prior triples	None	1 147 cpt., 21 rel.	Medium	OE
Thongkrau and Lalitrojwong (2012)	Latent Semantic Analysis	Prior Know.	Web documents	3 cpt.	Medium	OP
Vicient et al. (2013)	NER and Web-scale statistics	NER model	None	NA	Low	OE
Youn et al. (2020)	Domain specific embeddings	Wikip. corpus	None	100 cpt.	High	OP
Zhang et al. (2018)	Coupling CNN and RNN	Train. relations	Task specific	1 cpts, 4 rel.	High	OP

2.2.1. *Rule-based methods*

Rule-based extraction includes all the methods where a generic pattern is defined prior to the extraction and is built in order to extract a specific target. These patterns can be defined at different granularity levels and for different types of data sources. The main advantage of rule-based extraction methods is their high precision rate, due to the fact that each extraction rule is designed to extract a specific relation. Another advantage of rule-based methods is that they do not need any training dataset to get extracted instances, but only predefined rules. In return, these methods suffer from low recall performance since any data that does not match the predefined extraction rules will not be seen, even when they carry a strong hyponymy relation. Still, rule-based methods remain probably the most classical way to extract knowledge from data. The idea of these methods is to define a pattern which is an abstract representation of some kind of relation between terms. As listing covering rules for automated extraction from semi-structured data is possible, as proposed by Zhang and Li (2022) for XML documents, it is more difficult for unstructured data such as natural language. The complexity lies in the definition of this pattern as it should be generic enough in order to extract knowledge in large data sources and specific enough to avoid noise extraction that would match the pattern but will not carry any interesting knowledge.

Hearst (1992) defined a set of part-of-speech-based patterns to extract hyponymy (IS-A, or concept-instance) relations from raw text in order to enrich WordNet (Miller et al., 1990) with new semantic relations. Hearst's pattern consists of a sequence of part-of-speech tags that are to be found in a sentence as the expression of a hyponymy (concept-instance) relation. Textual data is then scanned with this pattern, and a relation is extracted each time the sequence is found. From these patterns, Pennacchiotti and Pantel (2006), but also De Silva and Jayaratne (2009) used instance extraction systems built on the rule-based approach to automatically extract domain-specific instances from raw text and Wikipedia articles, respectively. As mentioned above and initially highlighted by Hearst, one of the weaknesses of this method is the low recall. Language is indeed so fine and changing that it is impossible to find the exhaustive list of patterns that would describe a type of relation. Following this work, Herbelot and Copestake (2006) aimed at improving the extraction recall of Hearst patterns using Robust Minimal Recursion Semantics representation (Copestake et al., 2005; Copestake, 2006).

To push the limits of rule-based extraction methods a bit further, Pennacchiotti and Pantel (2006) also proposed a bootstrapping algorithm based on Hearst (1992)'s previous work, in order to iterate the process and extract new patterns from existing hyponymy relation extracted features. The idea of the algorithm is to take the two terms engaged in the previously detected hyponymy relation as a reference and to search for new occurrences of these terms in the data. Once some new occurrences are found, the pattern that linked these occurrences is extracted as a new pattern qualifying the hyponymy relation. This pattern can afterwards be used to extract new instances that are different from the first ones. The main difficulty of the bootstrapping method is that it still requires an initialization step. If the ontology remains completely unpopulated, there should be at least one pattern or one example of a related instance and concept per relation to initiate the algorithm. To address this limit, Nguyen et al. (2007) are proposing to start the process by automating the extraction of initial instances from Wikipedia. Also Toutanova et al. (2015) uses labelled text and existing knowledge base to find relational patterns and build an embedded representation of them through a convolutional neural network. Another aspect of the bootstrapping algorithm that can be limiting is the large number of rules, not necessarily relevant, that can be inferred during the bootstrapping step. To cope with this large number of rules, Ruiz-Casado et al. (2005) also proposed an algorithm to merge different patterns into a single pattern.

Kaushik and Chatterjee (2018), who worked on an agricultural ontology, proposed to extend pattern extraction techniques to a larger set of relations. Besides hyponymy relation (is-a), other relations defined within the ontology may have strong meanings. For instance, they defined a pattern detection method to extract *is-intercrop* relations, which express the capability of two agricultural species to grow in the same field. Although this pattern is very limited in terms of domain coverage, it still provides some guidance for defining generic patterns to extract non-taxonomic relations. Similarly, Alec (2023) defines a whole rule-based and knowledge-based process for ontology population from textual data. Despite the author's argument for a domain-independent approach in term of vocabulary, the proposed algorithm uses patterns that could not be reused for the extraction of other concept instances.

Rule-based methods are not new and have been widely used for instance detection tasks. Nevertheless, they remain to be a strong and precise method to extract knowledge from a document and are still supported by many natural language processing tools. More broadly, rule-based methods are still very much used for the extraction of different kinds of relations in various domains.

2.2.2. *Statistical and machine learning approaches*

Even if rule-based methods are considered as accurate methods, they cannot extract much information and require either some expertise to build the rules or already populated ontologies to instantiate the bootstrapping algorithm.

With the perspectives offered by the Internet of Things (Atzori et al., 2010), the development of linked open data (Auer et al., 2007) and the improvement of computing power, more and more studies are using statistical methods and machine learning driven approaches to tackle the problems of automatic knowledge extraction. In particular, the

development of efficient natural language processing tools allows better statistical analysis of textual data.

One way to invoke statistical analysis is to look at co-occurrences between words. De Boer et al. (2007) propose a method to extract relations involving components of an existing ontology. To choose which pairs of components should be linked, the authors study the statistical distribution of these instances within different documents and deduce which terms highly co-occur. This methodology showed good results when applied to deduce non-taxonomic relations between artists and musical genres. Similarly, Rajpathak (2013) extracts relations based on the co-occurrences between defective automobile pieces and actions taken.

A more global approach is taken by Thongkrau and Lalitrojwong (2012), who derive new instances of concepts from existing instances by representing the meaning of a term in a Latent Semantic Analysis (LSA) space. This LSA space, which represents the semantic meaning of a word in statistical terms, allows us to compute distances between this word and existing instances in order to deduce new concept-instance pairs by semantic similarity. These methods can also be referred to as close-world information extraction as they build models which suppose that every relation that they will be faced with is of the same nature as the one they have seen to build the statistical representation.

Not far from statistical approaches, the field of machine learning and deep learning carry many solutions for ontology population. Many studies use supervised learning to learn concepts and relations from ground true examples. Ferhat et al. (2022) couple ontology reasoning and machine learning to detect new defaults that are further proposed to an expert for the population of an existing knowledge base. This type of method, despite necessitating human intervention, can be assimilated to zero-shot learning, as new unseen defaults are extracted from real-world using existing knowledge but without prior occurrences of these defaults. Lomov et al. (2020) train a neural network to extend the set of concepts already identified within an ontology. The authors extract features from the context in which a term appears and train their model from these context features. Zhang et al. (2018) also use neural networks to extract drug-protein relations from medical reports. They combine recurrent neural network (RNN) and convolutional network (CNN) to handle both long and short sequences of data. The main limitation of these approaches is the need for annotated data or expert intervention since the algorithms used are trained in a supervised manner.

Besides the use of neural networks, the uprising of language models and word embedding participated in the emergence of new methodologies for relation extraction and ontology population. Ayadi et al. (2019) propose a method similar to Thongkrau and Lalitrojwong (2012)'s that assimilates an instance to a concept by comparing its representation within a word embedding to the representation of existing instance for which the concept is known already. This kind of approach is really common in the field and many studies try to take advantage of the ability of language models to represent the semantic aspect of words or even relations (Luo et al., 2020; Geng et al., 2020; Chen et al., 2018; Sanagavarapu et al., 2021; Huang et al., 2023). Such approaches deduce new relations from a large model. They can then be classified as open-world information extraction techniques, due to their ability to adapt to a wide range of domains of interest without retraining a model. Also, the use of a pre-trained language model has interesting advantages as it can be applied without further supervised training. Nevertheless, some of the approaches remain supervised since the language model is only used as a tool to generate semantic features representing an instance or a relation. The proper classification or relation detection algorithms remain supervised as they are trained from ground-true examples. However, this does not mean that unsupervised learning is never used for information extraction. Paukkeri et al. (2012) for example, use clustering algorithms recursively in order to build a taxonomy from previously extracted terms. Other methods, such as Youn et al. (2020) require the creation of domain-specific embeddings, supposing that domain-specific data is available for training.

2.2.3. Interest of a combined method

As explained in Sections 2.2.1 and 2.2.2, both approaches (rule-based and statistical) have their strengths and weaknesses. To overcome these limitations, some studies tend to use a hybrid strategy that combines both rule-based extraction and statistical or machine learning techniques. Kaushik and Chatterjee (2018) couple their rule-based approach for domain-related term identification with a statistical methodology in order to detect relations between these terms. Similarly, Alicante et al. (2016) use generic patterns to extract named entities from Italian text. These named entities are then enriched with additional contextual information. A clustering algorithm (k-means) is then applied to extracted entities in order to group potential relations. Torii et al. (2009) apply the hybridization methodology in reverse, first using statistical methods (hidden Markov model) to detect biomedical named entities before refining them with extraction rules. These studies show the interest of combining rule-based and statistical methods to improve the extraction process. Therefore, in the remainder of this article, the proposed approach is based on a combination of rule-based extraction and machine learning.

3. An hybrid generic framework for ontology population

3.1. Overall framework

The overall framework proposed in this work can be divided into two groups of processes. The first set of processes consists of parametric rules building, rule-based extraction, and data model building and form the rule-based process. This group of processes provides information and relations in order to instantiate the metamodel into a first data model. The rules are called parametric because their final form depends on the classes of the ontology, as presented in Section 3.2.1. These classes are also used to instantiate concepts of the data model.

The second set of processes includes Candidates extraction, Concept-candidate matching and Data model completion and builds up a semantic retroactive loop that leads to the registration of relations into the knowledge base through an alignment step with the initial classes of the ontology that has also been represented in the data model.

Once the rule-based process has been run on data and some hyponymy relations have been included in the ontology thanks to predefined parametric extraction rules, the semantic retroactive loop can be used. The initial data source is used more than once as it is needed for applying extraction rules and for exploring new relations based on validated hyponymy relations.

Fig. 1 provides a representation of the framework that has been described just before. In this article, Sections 3.2 through 3.3 focus on parametric rules building and rule-based extraction and Sections 3.4 and 3.5 focus on the semantic retroactive loop and associated concept-candidate semantic matching step.

3.2. Ontology-guided specific pattern extraction

The following sections detail the steps of construction of hyponymy relations extraction schemes from ontology concepts. Hearst patterns are presented, as well as the generalization method defined in this study and based on dependency parsing trees.

3.2.1. Patterns specification principle

A rule-based approach to information extraction requires the definition of domain-independent rules. In the specific case of text mining these rules take the form of extraction patterns. The difficulty in defining generic patterns, such as Hearst (1992)'s patterns is that they may identify a lot of hyponymy relations, even if the relation is not related to the domain. Using Hearst's patterns directly on a given text would then necessarily result in a lot of relations, among which a consequent number of relations will turn out to be irrelevant relatively to the ontology. Therefore, in order to limit the field of matched hyponymy

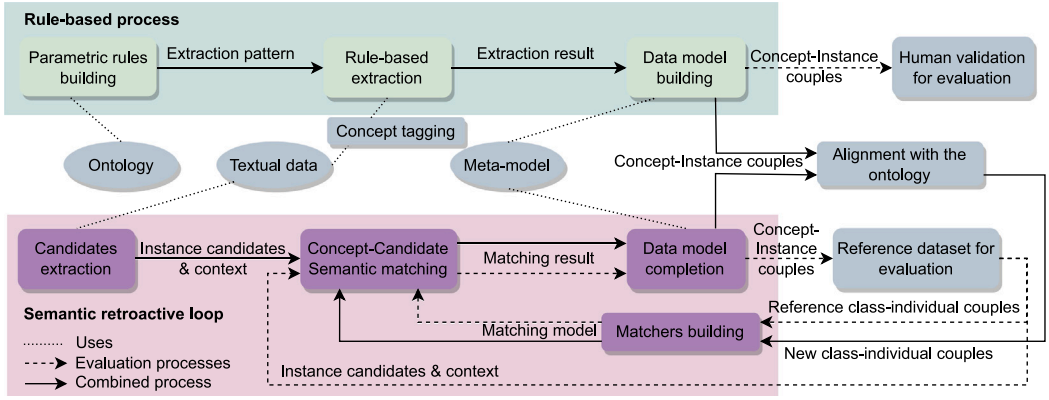


Fig. 1. Overall framework of the ontology population system.

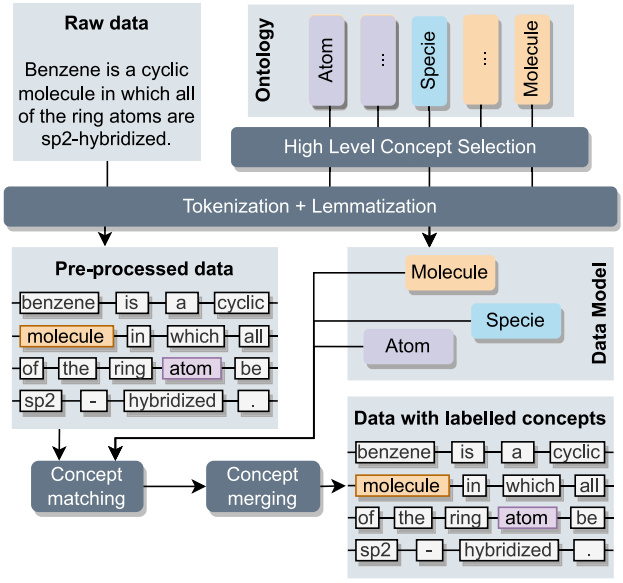


Fig. 2. Concept tagging process from ontology classes.

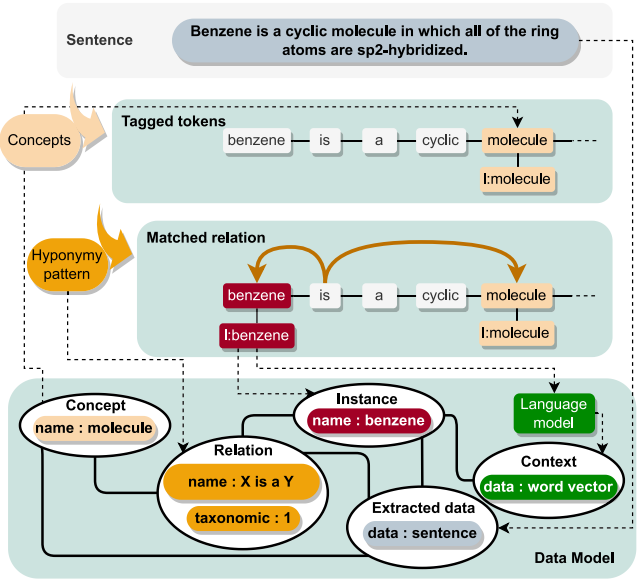


Fig. 3. Data model building process from raw data.

relations, the concepts of the ontology are used to specify generic Hearst's patterns so that they become domain-specific. This process of pattern specification can be adapted to any ontology as long as it contains concepts, what makes it generic.

3.2.2. Concept identification in data

The first step of the rule-based extraction task is to identify concepts in raw text based on their name, identified in the targeted ontology. For this purpose, classical natural language processing operations are used. Lemmatization is applied to the classes of the ontology. Obtained lemmas are used to create concepts in the data model. At the same time, the textual data is processed with tokenization and lemmatization. Then, a text matcher fed with the concept names is used on the preprocessed data to tag each term (or group of terms) as a concept if its lemma matches the concept. Fig. 2 summarizes this process.

3.2.3. Formal definition and application of patterns

In order to allow a reproducible application of patterns, a formal definition is needed. For this purpose, generic Hearst's patterns, defined by Chasseray et al. (2023), are applied to dependency parsing trees resulting from previous natural language processing steps. In Chasseray et al. (2023), such a pattern is defined by three sequences of restrictions. These sequences are called (1) part of speech sequence p, (2) dependency sequence d and (3) navigation sequence n. Elements of a p sequence are lists of possible part of speech tags that need to be encountered at each step. Similarly, an element of d sequence is a list of dependency tags that should link terms of the pattern having the correct part of speech tags. n sequence is used to indicate whether to seek next element of the pattern among the descendant (+1) or the ancestors (−1) of the current term. The combination of these three sequences draws the path that has to be matched between an identified concept and the targeted instance. Fig. 3 shows the application of the *I aux C* pattern to the example used earlier. The *I aux C* pattern is defined by the three following sequences : $p = [\ [aux], [prop, propn, noun]\ ]$, $d = [\ [attr], [nsubj]\ ]$ and $n = [-1,\ +1]$. When matched with the preprocessed sentence, this pattern allows to detect the hyponymy relation between the instance *benzene* and the concept *molecule*. The actions resulting from this detection are described in Section 3.3.

3.2.4. Handling mismatching patterns

The algorithm presented in Chasseray et al. (2023) is used to generically apply a pattern. Nevertheless, when patterns are applied straight as they are defined, some matching sequences may lead to incomplete – and consequently incorrect – extraction of information. As these mistakes can be corrected manually during a validation step, it may harm the automated aspect of the proposed framework.

Depending on the different nature of patterns, some additional processing has been added to (i) possibly extract several instances when

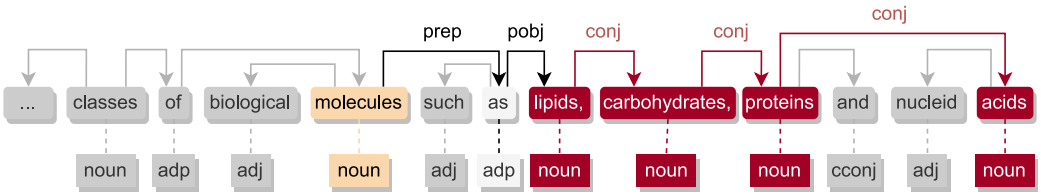


Fig. 4. Example of a post-processing step after the identification of an instance through *C pred I* pattern.

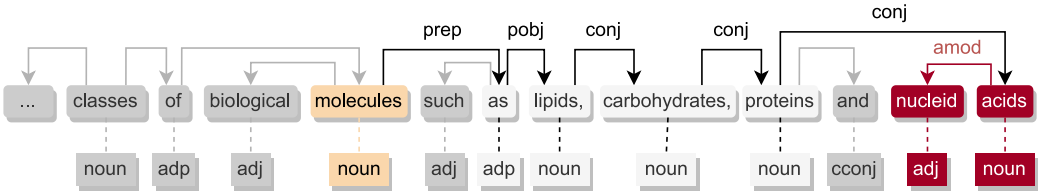


Fig. 5. Example of a post-processing step after the identification of an instance through *C pred I* pattern.

the pattern *C pred I* is used and (ii) complete matching instances by the pattern *C pred I*.

Detection of a sequence. The *C pred I* pattern has a particularity as it often identifies the first instance of a sequence of instances. Despite this first detected term is easily identified and registered as an instance, the following instances can be identified thanks to post-processing after applying the initial extraction pattern. To do so, an additional rule is applied to extract the remaining instances. This extraction rule starts from the first identified instance and recursively looks for a token having the following characteristics: $p = [prop, propn, noun]$, $d = [conj, adp]$ and $n = [+1]$, where p is the dependency linking the next instance to the already identified instances, d is the part of speech of the targeted token, and s is the direction in which the token is searched. Fig. 4 illustrates the application of this method through an example. In this example, a sequence of four instances is encountered. The post-processing procedure outlined in this section enables the identification of three additional terms, specifically *carbohydrate*, *protein*, and *acid*, as instances belonging to the concept of molecules.

Completion of instances. As a dependency tree is used to depict the syntactic relations occurring between terms, the identified term that supposedly reflects the targeted instance remains unique. In many cases, however, an instance can be expressed through a sequence of words. In order to recover this sequence, the extraction algorithm navigates the subtree of the targeted instance that has been detected by the pattern application algorithm and tries to find additional modifiers that complete the instance.

In the example presented in Fig. 4, the extracted instance, "acid", remains incomplete due to the omission of the *nucleid* modifier. Fig. 5 illustrates the application of the completion method to extract the last term of the sequence in its entirety (*nucleic acid*).

It is important to emphasize that while the example demonstrates instance completion within the context of sequence detection, this completion method can be applied universally to complete instances with modifiers in various patterns.

3.3. Information extraction and data model building

Once a pattern has matched a hyponymy relation in textual data, several pieces of information are extracted around this relation to extend the data model. The extracted elements concern the identified *Instance*, and the *Relation* that connects it to the already known *Concept*. The purpose of this part is to describe the elements that are extracted from raw data in order to create a data model according to the classes of the metamodel defined by Chasseray et al. (2021b).

3.3.1. Contextual elements

An identified instance is registered in the data model with the lemma of the corresponding tokens of the text. However, much additional contextual information can be extracted from raw data such as the frequency of the instance in text, the length of the instance in terms of tokens or characters, its position in the sentence, etc. In the generic metamodel for information extraction defined by Chasseray et al. (2021b), a Context class, is proposed in order to integrate this kind of information into the data model. It can be reused later to characterize, through learning mechanisms, instances that are more likely to be integrated into the ontology or refuted during human validation. This article focuses on a particular type of contextual information, which is the contextualized word vector built by applying a pre-trained BERT model to the instance related terms (Devlin et al., 2018). The embedding vector is built using the embedding layer of the transformer model. The extracted sentence in which the concerned instance has been identified is processed to merge all terms of the identified instance into one token. Then the transformer model is applied to the preprocessed sentence in order to build the embedding representation of the group of terms representing the instance. The same operation is also executed for the concept that is linked to the instance and has been merged during the process tagging step. This process ensures that different sentences, and then different extraction contexts lead to different embedding representations of both the instance and the concept.

3.3.2. Registering hyponymy relation

When the Instance is extracted and registered in the data model, the relation that links it to its concept must also be stored. Thus, each time a pattern detects a relation (implying an instance), the metamodel's Relation class is instantiated to associate the Instance lying in the data model with the concept from which it was detected. A relation is then defined with several attributes, which are :

- The name of the relation, directly deduced from the pattern that has been used to detect the relation.
- The subject concerned by relation, which, in the case of hyponymy relation, is the name of the extracted Instance.
- The object of the relation, which, in the case of hyponymy relations, is the name of the concept used to apply the pattern.

Extraction rules used in the case study presented in Section 4 are dedicated to hyponymy relation extraction. However, since ontologies can also have horizontal relations, patterns can be defined in order to detect relations that are not only taxonomic. Thus, another attribute is set that specifies whether a relation is taxonomic or not, depending on the rule used for extraction.

3.3.3. Extracted data class instantiation

The built model also contains instantiated versions of the Extracted data class in order to keep track of the raw data in the data model. Each time an instance is registered in the data model, the Extracted data class is instantiated with the sentence in which the instance has been found. Thus, the same instance can have multiple extracted data linked to it because it is found in several different sentences. Extracted data is then reused to support human validation and quickly contextualize instances.

3.4. Human validation

Since the patterns used remain relatively generic, the accuracy of the initial extraction remains uncertain. In order to evaluate the precision of the method it is necessary to involve an expert for a validation step. During this step, extracted concept-instance couples are presented to the expert in a contextualized manner, as they appear in textual data. As the same couple may have several occurrences in the same group of data, each of these occurrences is used as a contextualized version of the extracted couple in order to ease the decision. The domain expert can also provide further observations that can help identify the causal reason for an incorrectly extracted instance, or at least, signal an uncommon behaviour of the extraction patterns. 4 validation scenarios are proposed to the expert once a relation has been extracted, leading to 4 categories after validation:

- **Val**: The Concept-Instance relation is valid.
- **QVal**: The relation is correct but the involved Instance remains incomplete, incorrectly extracted or not specific enough to be considered as interesting in terms of knowledge intake.
- **Unc**: The extracted relation seems correct but the raw data is in contradiction with the expressed relation (wrong concept, unclear relation in the data).
- **Inc**: The Concept-Instance relation is not validated because declared incorrect, absurd or irrelevant by the domain expert.

The precision of the extraction is computed from *Val*, *QVal*, *Unc* and *Inv* categories distribution :

$$P = \frac{\%Val + \frac{\%QVal}{2}}{\%Val + \%Inc + \frac{\%QVal+\%Unc}{2}} \quad (1)$$

It is important to note the limited role of the human validation step, which is used to provide results to accurately assess the performances of the system. Its aim is not to guide the extraction, since the proposed system works in a fully automated manner. Compared with the previous steps (pattern extraction and data model building), validation is the more demanding step in terms of time, as each concept-instance couple requires 5 s for validation in average. However, the time taken by validation is not considered a limitation since the whole extraction system does not need the validation step to be executed.

3.5. Semantic retroactive loop

One of the characteristics of the framework is its iterative nature. This section is dedicated to the presentation of the semantic retroactive loop defined in the framework, which allows the discovery of new knowledge from previously validated relations. Relationships extracted using generic extraction rules can also be used as a support for the building of semantic extraction tools. This section gives details about the components used in the semantic retroactive loop.

While some instances fit the extraction patterns, a very large majority of them cannot be detected, simply because they do not appear explicitly near their associated concept in the explored data. As explained in the previous section, this represents a large amount of unrecognized knowledge. Nevertheless, the objective of this section is to extract these implicit instances and match them to the correct concept, based on context and previously validated instances. This is done through two main steps, which are (i) statistical candidate selection and (ii) candidate matching to ontology concepts. This section focuses specifically on the candidate matching step, while the candidate selection method is discussed in Section 5.1.

3.5.1. Use of word embeddings

Word embeddings, from the Word2Vec algorithm to transformers, make it easy to build a semantic representation of vocabulary. Transformers models have the advantage of semantically representing a term according to the context in which it occurs. The metamodel used to extract information from textual sources allows the definition of contextual information, through the context class. This context is then used to store a contextual vector derived directly from the instance's word embedding representation (as depicted in Fig. 3). Each vector is built by running RoBERTa pre trained transformers on the sentence from which the associated instance is extracted. This vector reflects the meaning of the instance as well as its context.

3.5.2. Matching process

In order to use the already extracted instances for the extraction of new instances among the statistically extracted candidates we introduce the concept of a matcher, whose role is to match a concept to a candidate when it makes sense to do so.

All defined matchers base their reasoning on a common hypothesis which assumes that two ontological objects share the same concept as long as they remain semantically close to each other. Thus, the objective of each matcher is to define a semantic distance between a candidate and a concept of the data model. The shortest distance between candidates and concepts leads to the construction of a matching relation. Depending on the matcher different methods are used :

- **WordNet similarity**: WordNet distance is computed thanks to the WordNet lexical database using the graph distance between the synsets including each term. The main limitation of this method is the lack of vocabulary in WordNet when it comes to specific and technical terms. WordNet distance principle is illustrated in Fig. 6 (left). The distance between two instances is calculated by taking the shortest path length between their respective WordNet synsets.
- **Word vector cosine similarity**: This measure uses the cosine similarity between instances' vectors to compute a similarity between instances and concepts. Word vector cosine similarity distance principle is illustrated in Fig. 6 (right). In certain scenarios, a single instance can manifest in multiple contexts, leading to several vectors representing it. In such instances, we calculate an average vector and subsequently compute distances based on this averaged representation.

The similarity between a candidate's name and a concept's name does not reveal the hyponymy relation that potentially links a candidate to a concept. Therefore, this similarity measure cannot be used directly between a candidate and a concept. However, the similarity between a candidate's name and an existing instance's name makes more sense. Then, to match a candidate with a concept, similarities are computed with the instance's names instead of the concepts' names. In Fig. 7, doted lines represent semantic similarities. In this example, Candidate 1 (Cand 1) presents higher semantic similarities with Instance 1 (Ins 1) and Instance 2 (Ins 2), whereas Candidate 2 (Cand 2) demonstrates a stronger semantic similarity with Instance 3 (Ins 3). Based on these similarity scores, it is more likely that Candidate 1 is associated with Concept 1 (Cpt 1), while Candidate 2 is linked to Concept 2 (Cpt 2).

A third matcher, illustrated in Fig. 8, uses the word vectors of validated instances to train a feed-forward neural network for probabilistic classification. Validated instances' vectors are used to train the classifier to pair each instance with the concept they are linked to in

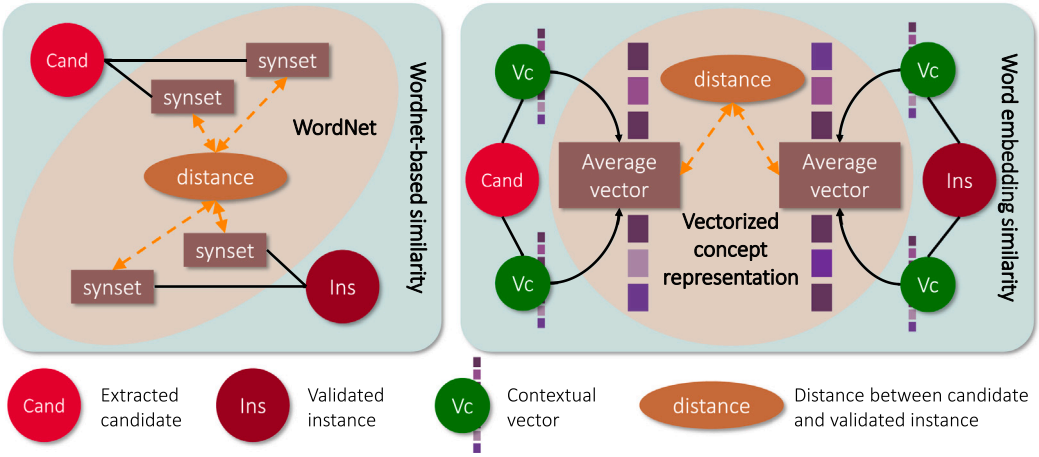


Fig. 6. Detailed representation of semantic similarities between a candidate and a validated instance.

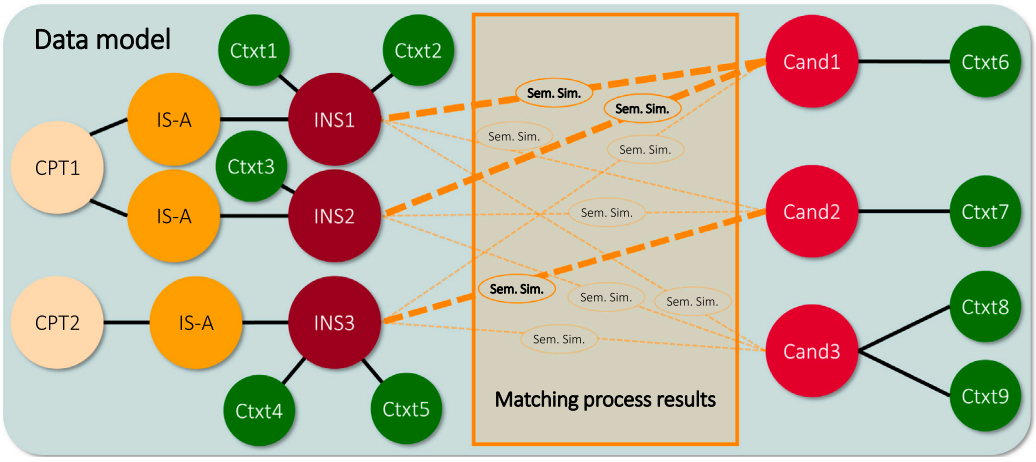


Fig. 7. Matching process between validated instances (left) and unmatched candidates (right) (Sem. Sim. : Semantic Similarity).

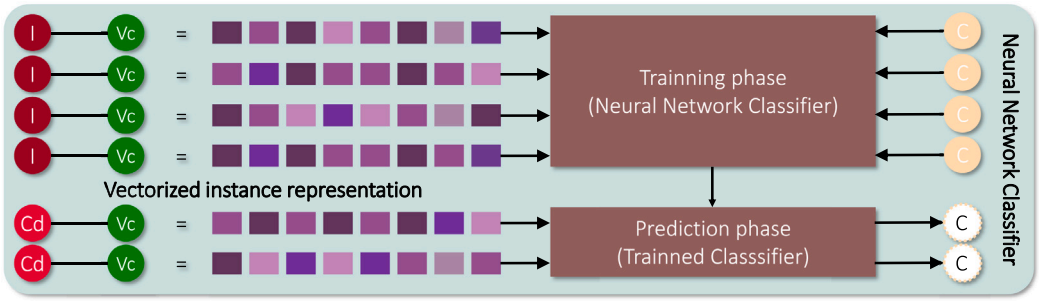


Fig. 8. Representation of the classifier matching process.

the data model. Word vectors of candidates are then used as features to determine to which ontology class the corresponding candidate should be assigned. The similarity between a candidate and a concept is then inferred from the probability of an instance being classified in the related class by the trained neural network.

3.6. *Technical architecture*

A technical implementation of the presented framework has been developed based on the components presented in Fig. 1 and listed in Fig. 9. In the presented architecture, the data model and an image of the ontology are stored thanks to a Neo4j graph database, which allows a graph representation of relations that is coherent with the definition of the information extraction metamodel and most ontology schema definitions. All data processing, including raw data pre-processing, natural language processing pipelines, candidates and relations extraction, alignment step and retroactive loop is provided by Python programming. Human validation is provided through a user interface built with a Django application. This user interface can also be used to launch extraction and retroactive loop.

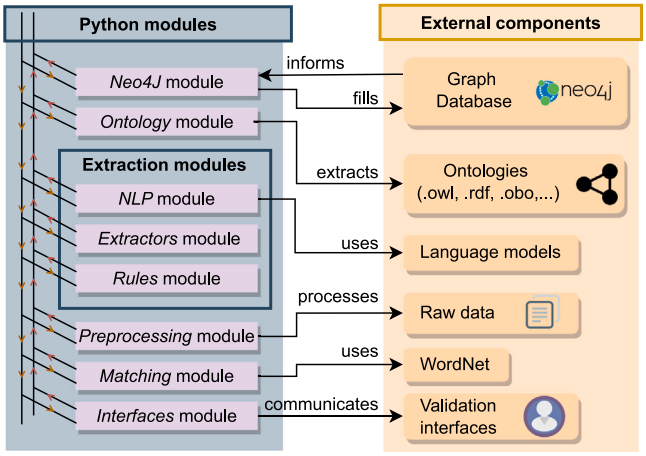


Fig. 9. Python modules of the technical implementation and their interactions with external components.

4. Application on textual data in the chemical and biochemical domain

4.1. Task distinction

The testing of the framework is divided into two groups of tasks. The goal is to test the pattern extraction pipeline and the semantic retroactive loop separately. Since the pattern extraction step is the first step of the pipeline, it can be directly applied and tested on textual data, as long as an ontology is available. The performance evaluation of this part of the framework is done by manual validation through the validation interface included in the prototype. On the other hand, the semantic retroactive loop assumes the existence of already extracted and validated instances that constitute a first knowledge base. In order to avoid approximation and bias due to the imperfection of pattern extraction and candidate selection, the semantic retroactive loop is tested on annotated data. A subset of this annotated data is used to simulate an initial knowledge base, while the rest of the data is used to test the semantic retroactive loop.

4.2. Selected ontologies

The first ontologies containing a significant number of classes (> 1 000) have been developed in the medical domain. In one of its first releases, the Unified Medical Language System gathered around 900,000 classes (Bodenreider, 2004). Later, and notably through the OBOFoundry (Smith et al., 2007), the number of available ontologies grew considerably and is covering today a wide range of domains in other related areas such as biomedical domain and biochemistry.

In this article, the prototype is applied to chemistry and biochemistry. Then, three ontologies, covering different aspects of these domains have been selected :

- **ChEBI (Chemical Entities of Biological Interest)**: ChEBI describes molecular elements that are relevant in the field of biology.
- **MOP (Molecular Process Ontology)**: The MOP ontology gathers a terminology for the description of molecular reactions between chemical entities.
- **RXNO (Name Reaction Ontology)**: This ontology can be considered as an extension of the two previous ontologies that detail organic interactions happening in the chemical processes described in the MOP ontology.

While the RXNO and MOP ontologies have several classes in common, they both differ from the ChEBI ontology in that they do not describe exactly the same domain. It was decided to use these three different ontologies to challenge the extraction system and analyse its ability to adapt to different sets of classes.

4.2.1. Gathering major concepts by splitting the ontology

Algorithm 1: HighLevelClassesSelection ($HLCS$)

Data: $classes$: classes of the ontology
$minNbBrchs$: minimal number of branches to consider the class as a high level class
Result: $nbSubBrch$: number of branches under the input classes
hlc : Set of high level classes of the ontology

```
begin
    nbBranches ⟵ 0
    hlc ⟵ ∅
    for c ∈ classes do
        subClasses ⟵ Descendants(c)
        nbSubBrchs, subHlc ⟵ HLCS(subClasses, minNbBrchs)
        for sc ∈ subClasses do
            if sc has descendants then
                nbBrchs ⟵ nbBrchs + 1
            else
                nbBrchs ⟵ nbBrchs + 1/2
        for shlc ∈ subHlc do
            extend(hlc, sHlc)
        if subNbBrchs > minNbBrchs then
            extend(hlc, c)
```

Some ontologies offer a very exhaustive level of granularity and can look like knowledge bases because they contain very precise concepts (e.g. Margherita pizza in the pedagogical pizza ontology). In order to retrieve concepts that have only a high level of granularity, one chooses to keep only the most generic classes of the ontology. This section describes the method used to estimate the *level of conceptualization at which a class is found* and thus recover only high-level classes to guide the extraction.

The algorithm 1 details how to select from the top classes of an ontology a subset of all classes of the ontology to guide an extraction, since they are considered to be high-level classes. Instead of defining the level of granularity from the top of the ontology, i.e. the most generic class, the algorithm looks at the number of finer classes associated with each class. Once this point of view is set, a class is considered important as soon as it is linked to enough finer classes. This threshold is set manually depending on the number of generic classes wanted. To determine the level of granularity for a class, a score of granularity is calculated using the following definition :

$$SG_c = W_c + \frac{1}{2} * |CF_c| + \sum_{cp \in CP_c} SG_{cp} \tag{2}$$

where :

- CP_c is the set of subclasses of c having no subclass itself.
- CF_c is the set of subclasses of c having itself at least one subclass.
- W_c is defined as follows :

$$W_c = \begin{cases} 1 & \text{if } c \text{ has at least one subclass} \\ 1/2 & \text{either} \end{cases} \tag{3}$$

4.3. Data presentation

This section is dedicated to the presentation of data used to test the proposed framework. The chosen ontologies for population are related to chemistry and biochemistry. One of the assumptions made in the building of the extraction framework is the coherence between

the ontology to be populated and the data that are processed. In fact, in order to make the concepts of the chosen ontology relevant for the pattern extraction process, it is necessary to choose textual data related to the same domain, i.e. in which instances of the classes of the ontology are likely to appear.

4.3.1. For the evaluation of the pattern extraction pipeline

In the presented framework, the pattern extraction step is fully unsupervised. Therefore, its performance should be estimated on raw and unlabelled text. For this purpose, several data sources have been selected. All of them contain textual data, presented in different formats, allowing to test the prototype on text with different levels of knowledge:

- Wikipedia articles related to pericyclic reactions
- Two different chemistry books in PDF format :
  - *Effects of Nanoconfinement on Catalysis* (Poli, 2017)
  - *The organometallic chemistry of the transition metals* (Crabtree, 2009)

4.3.2. For the evaluation of the semantic retroactive loop

Using the extracted dataset for the evaluation of the semantic retroactive loop seems to be relatively uncertain as the extracted couples are biased by the type of rule that has been used for their extraction. This aspect is discussed and detailed in the discussion section (Section 5). Moreover, in a classical use of the extraction system, these instances should all already be present in the data model since they have been extracted by a previous rule extraction. Therefore, it makes more sense to perform the semantic pairing and evaluate this pairing on a different kind of instances, that should not be extracted by the defined pattern.

In order to simulate such a situation – where a set of instances have been extracted and matched to a concept, and other (candidates) are waiting to be paired – the National Centre for Text Mining (NaCTeM) Dataset (Shardlow et al., 2018) has been used. This dataset is made of instances directly labelled from raw text and human annotations. It is used here for validation only, which does not mean that the framework needs annotated data to extract knowledge. It should be noted that the NaCTeM also includes the annotations of relations between labelled instances. As our dedicated task remains limited to the detection of hyponymy relations, only labelled instances from 6 different concepts (*Metabolite*, *Chemical*, *Protein*, *Specie*, *Biological activity* and *Spectral Data*) have been considered. The dataset have been preprocessed to avoid the consideration of other relations' labels so that only concept-instance relations are considered.

Concept-instance couples and associated semantic vectors are extracted from the annotated data and the associated textual sources. A subset of these couples leads to relations involving each associated concept in the data model. This method allows the generation of a pre-extracted and validated couples set (extracted set). Once the data model is filled with these couples, they can be used as a reference in order to match the remaining couples (candidates set) and thus evaluate the different pairing methods. The extracted set is used to train or guide the different matchers, while the candidate set is used to test these matchers on new instances.

4.4. Syntactic pattern extraction results

To apply extraction patterns the textual data is pre-processed in the following steps :

- **Tokenization** step during which each sentence of the text is split into several pieces.
- **POS-Taging** step applied to attribute a specific part-of-speech tag to each token, revealing its role in the sentence.

Table 2
Evaluation result on Poli (2017)'s document for three different ontologies.

Ontologies	Nextr	Neval	% Val	% Qval	% Inc	% Inv	P
ChEBI	929	452 (49%)	0.30	0.17	0.30	0.23	0.50
MOP	535	434 (81%)	0.42	0.16	0.26	0.17	0.62
RXNO	544	544 (100%)	0.37	0.14	0.28	0.22	0.55

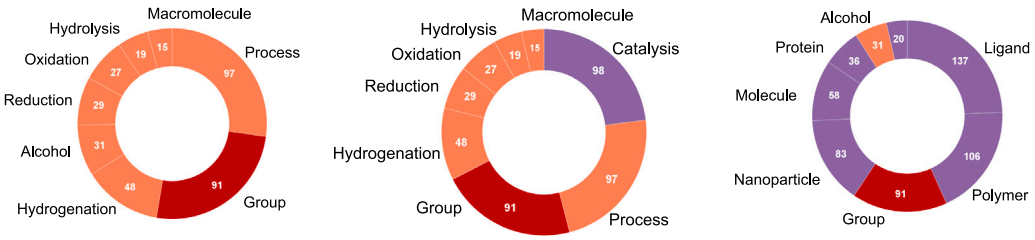
Table 3
Extraction results on the MOP ontology using the three different sources of data.

Source	Nextr	Neval	% Val	% Qval	% Inc	% Inv	P
Poli (2017)	535	434 (81%)	0.42	0.16	0.26	0.17	0.62
Crabtree (2009)	560	342 (61%)	0.41	0.15	0.21	0.23	0.62
Wiki articles	240	101 (42%)	0.47	0.13	0.20	0.21	0.63

- **Dependency parsing** step whose role is to deduce from part-of-speech tags, what syntactic relations each token has with the other tokens of the sentence.
- **Concept tagging** step based on selected concepts from the ontology as described in Section 3.2.2.

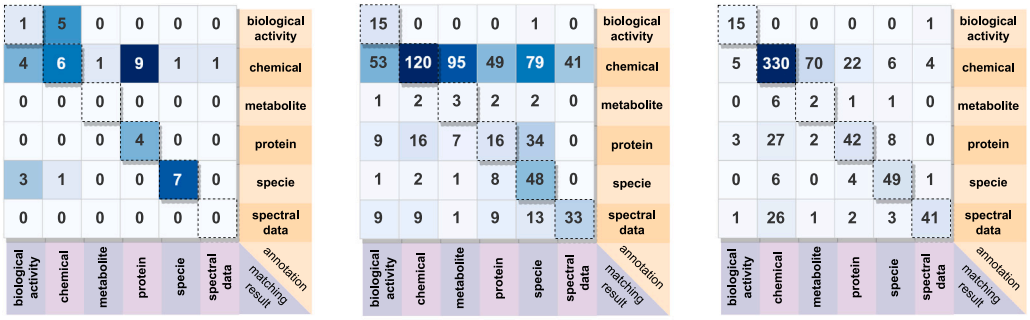
Fig. 10 shows the variability of selected concepts when processing the same pattern-based extraction with different domain ontologies. It can first be noticed that; within a single ontology, there is a disparity in terms of the number of concepts that are extracted. In MOP related extraction for example most of the instances concern the concepts *Process*, *Group* and *Catalysis*. It can be explained by the generic aspect of these concepts, whose instances are more likely to be found in all parts of a document about chemical knowledge than instances of more precise concepts such as *Oxydation*, *Reduction*, *Hydrolysis* or *Macromolecule*. A deeper analysis can be made by comparing the results on different ontologies. As explained in Section 4.2, MOP and RXNO describe the chemical domain with similar concepts. Due to this conceptual proximity, the relation extraction with the two ontologies involves globally the same concepts (*Macromolecule*, *Hydrolysis*, *Oxidation*, *Reduction*, *Process*). *Catalysis* instances, on the other hand, can only be found when using the MOP ontology, since this concept is only defined in the MOP ontology. For its part, the ChEBI ontology contains many concepts that cannot be found in the other two ontologies (*Polymer*, *Ligand*, *Protein*, *Mixture*, etc.). Thus, processing the same extraction with ChEBI ontology opens a wide new range of possible relations compared to MOP and RXNO ontologies. This shows the interest of the proposed method, which is able to focus specifically on the concepts defined on the ontology chosen for the extraction, without the need for manual adaptation. This is true within the same domain, for ontologies underlying different aspects of the domain, but also between completely distinct domains (Chasseray et al., 2021a).

In terms of performance, the Tables 2 and 3 show the results of human validation and the associated performance for different data sources and different ontologies. While the rules seem to perform slightly better on the MOP ontology, their performance remains very stable when applied to different documents. This observation confirms and completes the previous one, as the system can be applied to different ontologies and different textual documents without drastically affecting its performance. If we extrapolate the validation rate over the whole set of extracted relations, the global process results in the addition of 278 instances (ChEBI - Poli), 224 instances (MOP - Poli), 201 instances (RXNO - Poli), 229 instances (MOP - Crabtree) or 112 instances (MOP - Wikipedia) depending on the source of data and ontology considered. After the extraction process, the considered ontology is enriched with these instances and the taxonomic relations linking them to their corresponding concepts. Fig. 10 shows the repartition of these instances among the main concepts of the ontologies.



(a) RXNO (b) MOP (c) ChEBI

Fig. 10. Distribution of the 8 concepts leading to the most extracted relations for each ontology from Poli's (Poli, 2017) document. Concepts specific to one ontology (purple) are distinguished from concepts shared across 2 (orange) or 3 (red) ontologies.



(a) WordNet similarity (b) Word vector similarity (c) Neural Network training

Fig. 11. Confusion matrices built after the matching of candidates on annotated articles (each row indicates initial annotation which is compared to matching result (columns)).

4.5. Semantic loop performances

This section aims to prove the possibility of using a semantic retroactive loop to detect new instances from the relations extracted by syntactic patterns and validated by human validation. To accomplish this task, the three previously described methods – namely WordNet-based similarity, word-vectors cosine distance and feed-forward neural network – have been implemented. As described in Section 4.3.2, biochemical data are used for this application. The two datasets used contain 3 444 (ABS) and 17 207 (ART) concept-instance couples respectively. Non-taxonomic relations of the dataset have been filtered to only keep concept-instance couples. Once duplicated couples are removed the size of the datasets drops to 1493 (ABS) and 4 520 (ART). Both are divided into reference instances and test instances. The ABS test dataset represents 15% (678 instances) of the whole ART dataset whereas the ART test dataset represents 30% (448 instances) of the whole ABS dataset. The train datasets respectively contain the remaining 85% (3 842 instances) of ART dataset and 70% (1 045 instances) of ABS dataset. Figs. 11(a) to 11(c) show the resulting confusion matrices for the three different matching methods used on the two types of datasets. On these matrices, the diagonal coefficients represent the number of candidates that were correctly affected to their label, representing an ontology class. Table 4 gives a summarized view of the matching performances on both the ABS and ART datasets.

It can be noticed that the WordNet-based pairing method behaves poorly for almost every concept. As WordNet is a finite thesaurus, domain-specific terms are not necessarily included within this thesaurus. Thus, only a few candidate-instance pairs lead to the calculation of a distance based on WordNet. This explains why the number of classified candidates is lower when using the WordNet matcher. Nevertheless, *Protein* and *Specie* related performances are exceptions that confirm the previous idea as some species and proteins may have names that are common in the general speaking language and thus exist within WordNet. In fact, the WordNet matcher shows better precision for *Protein* instances because as soon as the instance exists in the WordNet thesaurus, it is relatively easy to match it to its concepts. Still, the recall remains low because many protein names are not referenced in WordNet. On the contrary, the WordNet matcher shows good recall performances for *Specie* instances, since most of the species (rat, human, mouse, apes, ...) are expressed with common words.

The similarity computed directly from word vectors shows a better performance for most of the concepts. However, the fact that the majority of the couples involve the concept *Chemical* results in a semantic ambiguity around this term. Because of this ambiguity, many chemical instances are incorrectly assigned to another concept. Using a neural network to build a model from the word vectors seems to reduce the semantic ambiguity, limiting it to the closest concepts only, such as *Chemical* and *Metabolite* for instance.

Another aspect is that the concepts are semantically close to each other and not completely exclusive, with some instances being paired with several concepts in the dataset. Since the classification allows multiple concepts but not multiple labelling, only the most represented annotation is kept for the building of the couples. Some instances present a classification that is considered incorrect even though they could have been labelled in this sense. For example, if a given instance has been labelled three times as a chemical but only once as a metabolite, the kept concept for this instance is chemical.

5. Discussion

This article addressed the issue of automated and domain-independent ontology population. This section discusses the limitations, applications and possible extensions of the proposed framework.

5.1. Statistical candidate selection

Matching candidates based on previously validated instances can only be assured if one has already extracted some candidates from data. Here, a candidate is defined as a word or a group of words representing an object identified in the data, that can later be identified as an instance of a given concept. Basically, the hypothesis is made that a candidate can be discovered among all the common nouns that can be

Table 4
Performance of the three different matchers computed for each concept of the labelled NaCTeM datasets.

	Biol. act.		Chemical		Metabolite		Protein		Specie		Spect. data		Global	
	P	R	P	R	P	R	P	R	P	R	P	R	P	R
ABS - WV	0.70	0.40	0.31	0.79	0.11	0.01	0.55	0.36	0.66	0.56	0.14	0.03	0.41	0.64
ART - WV	0.94	0.17	0.27	0.81	0.30	0.03	0.20	0.19	0.80	0.27	0.45	0.45	0.37	0.59
ABS - NN	0.73	0.71	0.75	0.77	0.33	0.05	0.25	0.55	0.64	0.78	0.86	0.60	0.66	0.72
ART - NN	0.94	0.63	0.76	0.84	0.20	0.03	0.51	0.59	0.82	0.73	0.55	0.87	0.72	0.76
ABS - WN	0.45	0.23	0.04	0.67	–	0.00	0.80	0.09	0.65	0.94	–	–	0.25	0.56
ART - WN	0.17	0.13	0.27	0.50	–	0.00	1.00	0.31	0.64	0.88	–	0.00	0.40	0.47

found in the textual data. However, since most of the common nouns in a technical text are not specifically related to the domain, there is a wide field of research that could be explored to orient the extractions towards topic-related terms.

In order to open new avenues in this field, we adapted the well-known TF-IDF measure for the extraction of relevant common nouns within a single document. The TF-IDF metric is used in order to enlighten terms of a document that are relevant, relatively to a corpus of documents. The limitation of TF-IDF is that it is not suitable for text that is not divided into several documents. Then, in order to detect terms of interest, a revised version of this metric could be investigated in order to be applicable to a single document.

5.2. Pattern-based extracted data bias

Section 4.3.2 suggests an underlying bias when using semantic similarity with previously pattern-based extracted couples for further annotation.

This is especially true for all the instances that include the name of the concept, which is the case for instances extracted by the *modifier* rule (*carbon atom* classified as an *atom*, for instance). The risk of having a significant amount of these kinds of instances is that their associated semantic vectors would only attract instances that include the name of their concept in their denomination, which is clearly only a small fraction of the instances that are interesting, even more so if they have already been extracted by a pattern. The underlying problem is that if the initially extracted instances are not sufficiently representative of the variety of instances to which a concept could be associated, the system may have difficulty in associating new candidates with these concepts. This applies to the morphology of the extracted instances but also concerns their number. It should be difficult to correctly assign new instances to a concept if that concept is initially associated with a small number of instances and associated semantic vectors. It also seems that the concepts that allow the most instances to be extracted by pattern application are some of the most generic concepts of the ontology. This fact means that further extraction is likely to favour these concepts, limiting the possibilities for more specific concepts to be associated with new instances since their chances of being associated are null if no instance has been previously extracted for the former concept.

5.3. Interest in targeted document analysis

A direct application of a system like the one proposed in this article concerns the exploration of a domain with self-defined concepts. For example, in the chemical domain, where many terms may be specific to a single reaction or reaction step, using the extraction system with precise concepts can be a method to get small interesting extracts of a document that contain instances of the given concept and that would not have been extracted with other methods (using keyword-based extraction for example).

5.4. Limits inherent to the unsupervised nature of the approach

All of these three matching methods present a major limitation due to made hypothesis that semantic relations between a candidate and instances mean that this candidate belongs to the same concept as the instance. Matchers, as they tend to match a concept for candidates could match candidates that should not as they have no corresponding concept in the ontology. Matching performance is then really impacted by the quality of the extracted candidates. Yet, another condition needed to guarantee good performance in the matching step is to have candidates that are instances of at least one of the classes of the ontology. Still, this is a challenge when adopting an unsupervised and domain-free method.

5.5. Domain related limits

In this article, the methodology has been applied to three ontologies about the chemical domain. It has been noticed that the classes of the ontology that are more likely to be populated are relatively generic regarding the domain (molecule, process, alcohol, protein, etc.). It is then possible that the used patterns are not designed for more specific classes, whose instances do not appear close to the class occurrence. It can then legitimately be asked if more specific ontologies about technical domains may limit the extraction capacity of patterns.

5.6. Extension to other data sources and other relations

The proposed framework and method have been implemented for textual data and extraction of hyponymy relations. However, ontologies not only convey taxonomic relations but can express other types of relations. The same framework can nevertheless be used to cover different types of data (speech, image, structured text) or to search for other types of relations. Future work on this topic could then be to define the patterns or rules that can be used to treat new types of either sources or relations. Searching for relations could then be a way to overcome the specific domain limitation as relations may link classes which are not efficiently populated through hyponymy patterns, but would be through other types of relation patterns.

6. Conclusion

This article presented a complete framework for unsupervised ontology population. Several blocks of the framework have been detailed, from high concept selection in the targeted ontology to rule-based extraction mechanism and semantic retroactive loop matching process. These blocks have been implemented and showed good results that (1) prove the possibility of extracting instances for several types of documents and different domain ontologies with reasonable impact on the performance and (2) show the possibility of enriching a knowledge base within the framework by using already extracted instances to discover new ones. Some future work has been identified with the objective of extending the framework to other types of data and relations. Limitations concerning the usage of generic word embeddings has also been evoked opening new perspectives on the training of domain-specific word embeddings for better semantic matching on specific domains.

Funding sources

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

CRediT authorship contribution statement

Yohann Chasseray: Writing – review & editing, Writing – original draft, Validation, Project administration, Methodology, Data curation, Conceptualization. **Anne-Marie Barthe-Delanoë:** Writing – review & editing, Validation, Supervision, Project administration, Funding acquisition. **Jérome Volkman:** Writing – review & editing, Validation, Data curation, Conceptualization. **Stéphane Négny:** Writing – review & editing, Supervision, Project administration. **Jean Marc Le Lann:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

A link to a git repository containing used data is shared in the article.

Appendix A. Supplementary material

A list of Wikipedia articles on pericyclic reactions and dataset of extracted and validated concept instance couples are available on the following git repository: https://gitlab.mines-albi.fr/ychasser/adopi-data.

References

Alec, C., 2023. Ontology population from french classified ads. In: International Conference on Conceptual Structures. Springer, pp. 155–170.

Alicante, A., Corazza, A., Isgrò, F., Silvestri, S., 2016. Unsupervised entity and relation extraction from clinical records in Italian. Comput. Biol. Med. 72, 263–275.

Arnold, P., Rahm, E., 2014. Enriching ontology mappings with semantic relations. Data Knowl. Eng. 93, 1–18.

Atzori, L., Iera, A., Morabito, G., 2010. The internet of things: A survey. Comput. Netw. 54 (15), 2787–2805.

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z., 2007. Dbpedia: A nucleus for a web of open data. In: The Semantic Web. Springer, pp. 722–735.

Ayadi, A., Samet, A., de Beuvron, F.d.B., Zanni-Merk, C., 2019. Ontology population with deep learning-based NLP: a case study on the biomolecular network ontology. Procedia Comput. Sci. 159, 572–581.

Bodenreider, O., 2004. The unified medical language system (UMLS): integrating biomedical terminology. Nucleic Acids Res. 32 (Suppl. 1), D267–D270.

Bosselut, A., Rashkin, H., Sap, M., Malaviya, C., Celikyilmaz, A., Choi, Y., 2019. COMET: Commonsense transformers for automatic knowledge graph construction. arXiv preprint arXiv:1906.05317.

Chasseray, Y., Barthe-Delanoë, A.M., Négny, S., Le Lann, J.M., 2021a. Automated unsupervised ontology population system applied to crisis management domain. In: ISCRAM 2021-18th International Conference on Information Systems for Crisis Response and Management. (2389), pp. p–968.

Chasseray, Y., Barthe-Delanoë, A.-M., Négny, S., Le Lann, J.M., 2021b. A generic metamodel for data extraction and generic ontology population. J. Inf. Sci. 0165551521989641.

Chasseray, Y., Barthe-Delanoë, A.-M., Négny, S., Le Lann, J.M., 2023. Knowledge extraction from textual data and performance evaluation in an unsupervised context. Inform. Sci. 629, 324–343.

Chen, M., Tian, Y., Chen, X., Xue, Z., Zaniolo, C., 2018. On2vec: Embedding-based relation prediction for ontology population. In: Proceedings of the 2018 SIAM International Conference on Data Mining. SIAM, pp. 315–323.

Copestake, A., 2006. Robust minimal recursion semantics. Technical Report, Cambridge Computer Lab. Unpublished.

Copestake, A., Flickinger, D., Pollard, C., Sag, I.A., 2005. Minimal recursion semantics: An introduction. Res. Lang. Comput. 3 (2–3), 281–332.

Crabtree, R.H., 2009. The organometallic chemistry of the transition metals. John Wiley & Sons.

De Boer, V., van Someren, M., Wielinga, B.J., 2007. A redundancy-based method for the extraction of relation instances from the web. Int. J. Hum.-Comput. Stud. 65 (9), 816–831.

De Silva, L., Jayaratne, L., 2009. Semi-automatic extraction and modeling of ontologies using wikipedia XML corpus. In: 2009 Second International Conference on the Applications of Digital Information and Web Technologies. IEEE, pp. 446–451.

Devlin, J., Chang, M.W., Lee, K., Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Ferhat, M., Leray, P., Ritou, M., Le Du, N., 2022. Iterative knowledge discovery for fault detection in manufacturing systems. Procedia Comput. Sci. 207, 744–753.

Geng, Z., Chen, G., Han, Y., Lu, G., Li, F., 2020. Semantic relation extraction using sequential and tree-structured lstm with attention. Inform. Sci. 509.

Gruber, T.R., 1993. A translation approach to portable ontology specifications. Knowl. Acquis. 5 (2), 199–220.

Hearst, M.A., 1992. Automatic acquisition of hyponyms from large text corpora. In: Proceedings of the 14th Conference on Computational Linguistics, vol. 2, Association for Computational Linguistics, pp. 539–545.

Herbelot, A., Copestake, A., 2006. Acquiring ontological relationships from wikipedia using rmrs. In: Proc. ISWC 2006 Workshop on Web Content Mining with Human Language Technologies.

Huang, H., Yuan, C., Liu, Q., Cao, Y., 2023. Document-level relation extraction via separate relation representation and logical reasoning. ACM Trans. Inf. Syst. 42 (1), http://dx.doi.org/10.1145/3597610.

Kaushik, N., Chatterjee, N., 2018. Automatic relationship extraction from agricultural text for ontology construction. Inf. Process. Agricult. 5 (1), 60–73.

Khadir, A.C., Aliane, H., Guessoum, A., 2021. Ontology learning: Grand tour and challenges. Comp. Sci. Rev. 39, 100339.

Konys, A., 2018. Knowledge systematization for ontology learning methods. Procedia Comput. Sci. 126, 2194–2207.

Konys, A., 2022. An ontology-based approach for knowledge acquisition: An example of sustainable supplier selection domain corpus. Electronics 11 (23), 4012.

Leshcheva, I., Begler, A., 2022. A method of semi-automated ontology population from multiple semi-structured data sources. J. Inf. Sci. 48 (2), 223–236.

Lomov, P., Malozemova, M., Shishaev, M., 2020. Training and application of neural-network language model for ontology population. In: Proceedings of the Computational Methods in Systems and Software. Springer, pp. 919–926.

Louge, T., Karray, M.H., Archimède, B., 2018. Investigating a method for automatic construction and population of ontologies for services: performances and limitations. In: 2018 IEEE/ACS 15th International Conference on Computer Systems and Applications. AICCSA, IEEE, pp. 1–6.

Luo, L., Yang, Z., Cao, M., Wang, L., Zhang, Y., Lin, H., 2020. A neural network-based joint learning approach for biomedical entity and relation extraction from biomedical literature. J. Biomed. Inform. 103.

Miller, G.A., Beckwith, R., Fellbaum, C., Gross, D., Miller, K.J., 1990. Introduction to WordNet: An on-line lexical database. Inn J. Lexicogr. 3 (4), 235–244.

Nguyen, D.P., Matsuo, Y., Ishizuka, M., 2007. Exploiting syntactic and semantic information for relation extraction from wikipedia. In: IJCAI Workshop on Text-Mining & Link-Analysis (TextLink 2007).

Nicola, A.D., Missikoff, M., Navigli, R., 2009. A software engineering approach to ontology building. Inf. Syst. 34 (2), 258–275. http://dx.doi.org/10.1016/j.is.2008.07.002, URL: http://www.sciencedirect.com/science/article/pii/S0306437908000628.

Paukkeri, M.S., García-Plaza, A.P., Fresno, V., Unanue, R.M., Honkela, T., 2012. Learning a taxonomy from a set of text documents. Appl. Soft Comput. 12 (3), 1138–1148.

Pennacchiotti, M., Pantel, P., 2006. A bootstrapping algorithm for automatically harvesting semantic relations. In: Proceedings of the Fifth International Workshop on Inference in Computational Semantics. ICoS-5, p. 87.

Poli, R., 2017. Effects of nanoconfinement on catalysis. Springer.

Rajpathak, D.G., 2013. An ontology based text mining system for knowledge discovery from the diagnosis data in the automotive domain. Comput. Ind. 64 (5), 565–580.

Rajput, A.M., Gurulingappa, H., 2013. Semi-automatic approach for ontology enrichment using umls. Procedia Comput. Sci. 23, 78–83.

Rani, M., Dhar, A.K., Vyas, O., 2017. Semi-automatic terminology ontology learning based on topic modeling. Eng. Appl. Artif. Intell. 63, 108–125.

Ruiz-Casado, M., Alfonseca, E., Castells, P., 2005. Automatic extraction of semantic relationships for wordnet by means of pattern learning from wikipedia. In: International Conference on Application of Natural Language To Information Systems. Springer, pp. 67–79.

Sanagavarapu, L.M., Iyer, V., Reddy, Y.R., 2021. OntoEnricher: A deep learning approach for ontology enrichment from unstructured text. arXiv preprint arXiv:2102.04081.

Shardlow, M., Nguyen, N., Owen, G., O'Donovan, C., Leach, A., McNaught, J., Turner, S., Ananiadou, S., 2018. A new corpus to support text mining for the curation of metabolites in the chebi database. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation. LREC 2018.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L.J., Eilbeck, K., Ireland, A., Mungall, C.J., et al., 2007. The OBO foundry: coordinated evolution of ontologies to support biomedical data integration. Nature Biotechnol. 25 (11), 1251–1255.

Thongkrau, T., Lalitrojwong, P., 2012. Ontopop: An ontology population system for the semantic web. IEICE Trans. Inf. Syst. 95 (4), 921–931.
Torii, M., Hu, Z., Wu, C.H., Liu, H., 2009. BioTagger-GM: a gene/protein name recognition system. J. Am. Med. Inform. Assoc. 16 (2), 247–255.
Toutanova, K., Chen, D., Pantel, P., Poon, H., Choudhury, P., Gamon, M., 2015. Representing text for joint embedding of text and knowledge bases. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. pp. 1499–1509.
Vicient, C., Sánchez, D., Moreno, A., 2013. An automatic approach for ontology-based feature extraction from heterogeneous textualresources. Eng. Appl. Artif. Intell. 26 (3), 1092–1106.
Youn, J., Naravane, T., Tagkopoulos, I., 2020. Using word embeddings to learn a better food ontology. Frontiers Artif. Intell. 3, 584784.
Zhang, F., Li, Q., 2022. Constructing ontologies by mining deep semantics from XML schemas and XML instance documents. Int. J. Intell. Syst. 37 (1), 661–698.
Zhang, Y., Lin, H., Yang, Z., Wang, J., Zhang, S., Sun, Y., Yang, L., 2018. A hybrid model based on neural networks for biomedical relation extraction. J. Biomed. Inform. 81, 83–92.