Knowledge extraction from textual data and performance evaluation in an unsupervised context

Yohann Chasseray, Anne-Marie Barthe-Delanoë, Stéphane Négny, Jean-Marc Le Lann

Knowledge extraction from textual data and performance evaluation in an unsupervised context

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

[a] *Laboratoire de Génie Chimique, Université de Toulouse, CNRS, INPT, UPS, Toulouse, France*
[b] *Centre Génie Industriel, Université de Toulouse, IMT Mines Albi, France*

A B S T R A C T

Among the incoming challenges in monitoring systems, the aggregation, synthesis and management of knowledge through ontological structures hold an essential place. Existing knowledge extraction systems often use a supervised approach that relies on annotated data, inducing implicitly a fastidious annotation process. Current research is towards the definition of unsupervised or semi-supervised systems, allowing a wider range of knowledge extraction. The evaluation of such systems, performing knowledge extraction using natural language processing methods requires performance indicators. The indicators usually used in such evaluations have limitations in the specific context of knowledge extraction for unsupervised ontology population. Thus, the definition of new evaluation methods becomes a need arising from the singularity of the harvested data, especially when these are not annotated. Hence, this article proposes a method for measuring performance in unsupervised context where reference data and extracted data do not overlap optimally. The proposed evaluation method is based on the exploitation of data that serve as a reference but are not specifically linked to the data used for extraction, which makes it an original evaluation method. To apply the performance measure on concrete cases, this paper also presents an unsupervised self-feeding rule-based approach for domain-independent ontology population from textual data.

1. Introduction

Building and automating systems requires the support of a solid structure based on real-time perception of the internal and external context in which the given system evolves. In the case of a production line, it remains essential to keep detailed knowledge or at least to have an overview of downstream demand and upstream resources, in order to foresee or deal with any overstocking or downtime in the production. Based on this observation, decision support systems allow not only to represent the state of production systems but also to create a predictive analysis of near future system behavior (demand forecasting, predictive maintenance). The current trend in the development of such systems and tools is to increase the addition of human-based knowledge. Adding this knowledge allows decision support systems to adopt a contextualized way of reasoning which is more and more similar to human reasoning abilities and thus to propose better support in the decision-making process. The incorporation of knowledge to decision

* Corresponding author.
E-mail addresses: yohann.chasseray@mines-albi.fr (Y. Chasseray), anne-marie.barthe@mines-albi.fr (A.-M. Barthe-Delanoë), stephane.negny@ensiacet.fr (S. Négny), jeanmarc.lelann@ensiacet.fr (J.-M. Le Lann).

support systems is materialized by the integration of knowledge bases, that can be of different types such as knowledge graphs [21] or queried databases [14] for instance. However, more and more knowledge bases are obtained through the extension of an ontology developed for the domain of interest. Ontologies, initially defined by Gruber [17] as *the explicit specification of a conceptualization*, constitute an important support for the representation of knowledge within a domain. Whereas ontologies provide general knowledge about the main concepts of a domain, a knowledge base can be defined as the instantiated version of an ontology in the specific context of its application to a defined problem. A knowledge base thus includes a set of instances derived from the concepts of the ontology and a set of relations between these instances, derived from the relations defined at the level of the ontology.

Unfortunately, and despite the growing adoption of ontology-based approaches, most of the developed systems work with an ontology specific to the application cases, which leads to knowledge bases that can only be used in those specific cases. Indeed, the majority of ontologies and knowledge bases are mostly built following a strategy proper to the problem to solve making their re-use limited or even impossible. Yet more generic ontologies exist towards. However, the knowledge bases built on these ontologies, when they exist, contain partial or little knowledge at the application level, making their use also very limited. At the same time, human-generated data, in more or less structured formats, remain very rich in terms of the knowledge they contain. Meanwhile, one assists to the emergence of unstructured data processing and analysis techniques which represent significant opportunities for the automatic population of ontologies considering the increasing flow of unstructured data. Numerous methodologies have been proposed to explore these opportunities in the literature. However, just a few of them focus their work on the evaluation task of such systems. Mainly, the evaluation of extraction frameworks is processed through a comparison of extracted relations with a gold standard, restricting validation to a binary decision (1 if a piece of information is detected by the system, 0 when the system missed it). Then, when dealing with unstructured text, a gold standard only represents a unique version of extracted information. This kind of evaluation does not take into account all the instances that are partially retrieved, or information that was not supposed to be retrieved if one follows the gold standard but convey knowledge that should not be avoided.

This observation opens many new objectives concerning the gathering of knowledge and information. Considering it, this paper presents the result of research work that has been conducted integrating the three following objectives:

- Investigating new methods of unsupervised ontology population by taking advantage of the targeted ontology but without previous knowledge about the domain of application related to this ontology.
- Tackling the difficulties implied by the unsupervised character of the proposed extraction methods, especially in terms of model evaluation.
- Ensuring the integration of the proposed methods in a wider framework so that they can automatically be reproduced with any ontology, covering any domain.

The goal of the presented work is to answer these challenges and provide some perspectives towards the automation of unsupervised knowledge extraction for ontology population. To reach this objective, two contributions are detailed in this paper. First, a domain independent rule-based method for unsupervised relation extraction is presented as a part of a wider framework for ontology population. The second contribution is a generic and flexible approach for automated evaluation of an ontology population system based on an existing dataset of concept-instance couples. This reference dataset can be built either from gold standard datasets or from existing knowledge bases as its structure remains simple, making the method reproductible for several types of reference data. Both contributions have been applied to two different sets of textual data, in order to show their re-usability on different domains and different sources of reference data.

This article, which discusses different aspects of building knowledge bases from pre-existing domain ontologies, is organized as follows: section 2 presents a state-of-the-art of the methods used for ontology population and more precisely for relation extraction. Existing performance measures and their limitations in unsupervised context are also discussed further. Section 3 introduces a generic framework for the domain-independent ontology population task and takes a closer look at the specification of this framework for knowledge extraction from textual data. Section 4 details a method for extracting hyponymy relations (concept-instance) based on the use of extraction patterns and existing knowledge carried by the ontology. Finally, section 5 constitutes a step back from this framework and proposes a method for the measurement of performance of such an extraction system. Before drawing conclusion, the proposed method for the evaluation of an extraction system is discussed in section 6.

2. State of the art

2.1. Ontology population

Ontology population issues are not new and several studies have already focused on the exploitation of raw data in order to build a knowledge base. In this study, ontology population is distinguished from ontology learning which tends to learn new representation of knowledge instead of using an existing one.

The mostly used methods for ontology learning and ontology population can be classified in three kinds of approaches:

- **Rule-based methods** in which patterns representing relations are defined and applied to unstructured data in order to detect these relations.
- **Statistical methods** looking at data more globally and taking advantage of the volume to derive semantic information from unstructured data.

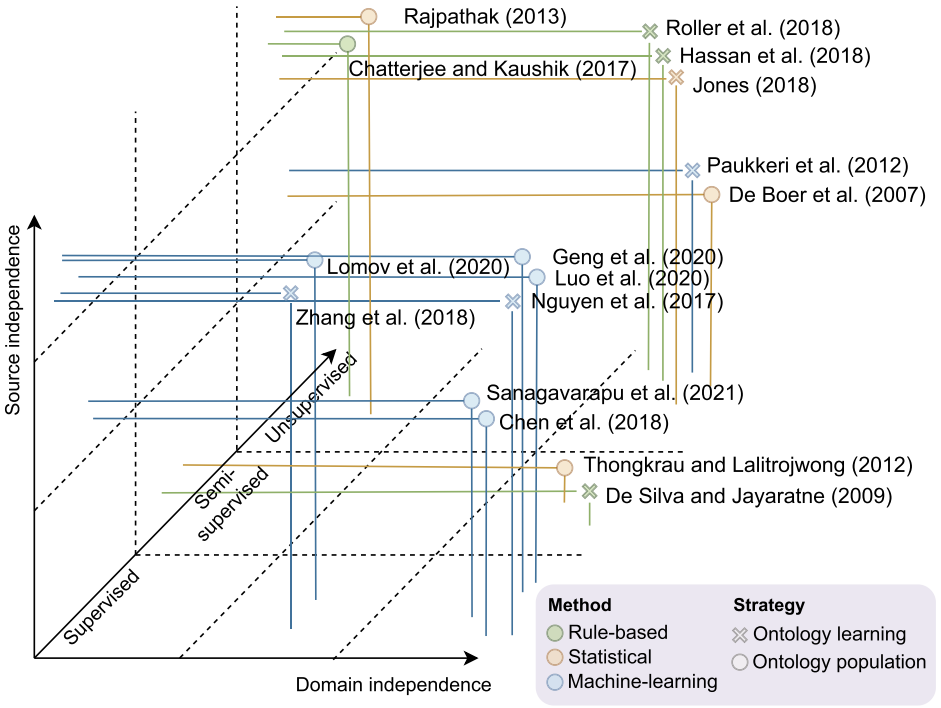

Fig. 1. Evaluation of literature approaches for ontology population/learning.

- **Machine learning and deep learning based methods** using annotated data or pre-trained models in order to learn relations within a given piece of unstructured data.

Each approach will be analyzed regarding three criteria which are (i) the need of genericity regarding the domain of application, (ii) the ability to process different sources of data and (iii) the possibility to be processed without reference data and in an unsupervised way. The results of the evaluation of the different approaches are presented in Fig. 1.

2.1.1. Rule-based methods

Rule-based methods are unsupervised methods that define rules in advance. These rules are representative of the relation that one wants to detect. Some of these methods can still be very restricted to the domain of application or to the source of data depending on how the rules are defined.

For instance Chatterjee and Kaushik [7] developed an algorithm (RENT) based on extraction rules and regular expression to extract hyponymy relations in the agricultural domain. Despite the algorithm shows good performances in terms of precision (80%) and recall (60%),[1] one of its main limitations is that it is restricted to agricultural domain.

WikiOnto is a system developed by De Silva and Jayaratne [10] in order to extract concepts from XML (eXtended Markup Language) documents of the Wikipedia encyclopedia. This system applies rules at different levels (XML structure, lexical level, grammar level) and can adapt to different domains of application as long as it sticks to the structure of Wikipedia XML Data.

Hearst [20] defined generic patterns that can be applied to any textual data and in any domain. Hassan et al. [19] and Roller et al. [35] integrated Hearst's patterns in a larger framework and applied them on a cross-domain evaluation dataset and different hypernym[2] detection tasks, showing good average precision results. Despite Hearst's patterns generally show good precision performance keeping user implication low, their application is only retrieving a fraction of the knowledge that is contained in data sources as patterns focus on most frequently appearing expressions of a relation.

2.1.2. Statistical methods

Rule-based methods are mainly focused on small parts of data, as rules are generally applied between instances that are close to each other. Statistical methods, on the contrary, provide a global insight of the relevant terms that can be extracted from large corpora. Hence they are also used in order to guide ontology population.

[1] Precision is computed comparing good extractions of the system with wrongly extracted elements. Recall is obtained comparing the amount of effectively detected elements with the global amount of elements that should have been detected.

[2] An hypernym designates an instance that is involved in a hyponymy – or concept-instance – relation.

Co-occurrence analysis Stating that the closer terms appear from each other, the more likely they are to be linked by a relation, many studies oriented their work towards the analysis of words co-occurrences. De Boer et al. [9] based their extraction system on co-occurrence analysis. They used Web extracted documents to deduce relations from existing instances and based on how often they appear next to each other allowing a simple and quick search of occurrences of predefined relations.

Detection of relevant terms Some methods are useful for ontology population as they allow the extraction of terms that appear to be relevant regarding the subject. TF-IDF [23] is one of them as this method weights terms of a document comparing their appearance rate in a document with their appearance rate in a corpus of documents. Despite it is not able to link them to an ontology, this method allows the extraction of the most relevant terms of a document.

Latent Semantic Analysis Latent Semantic Analysis (LSA) is a statistical method used initially to classify documents from the vocabulary they use [26]. Thongkrau and Lalitrojwong [39] adapted LSA to the task of hypernym detection. Using already existing examples, the authors computed a LSA Space and used its semantic properties to deduce new relations. One of the limitations of this method is its high dependency regarding the kind of textual data which is processed.

2.1.3. Machine learning and deep learning based methods

Machine learning and deep learning provide solutions for ontology population as well. Clustering methods and neural network training are commonly used for ontology learning and relation detection.

Automated ontology learning Some approaches use clustering methods to group terms into subsets, in order to learn the concepts of an ontology [33,15,29,37]. These methods present the advantage of being unsupervised but learn an ontology from the ground up without populating an existing one. For instance, Paukkeri et al. [33] use hierarchical clustering to deduce a taxonomy from Wikipedia entries. Nevertheless, the built taxonomy is not linked to any existing ontology. De Silva and Jayaratne [10] use k-means algorithm to group the keywords of a document based on their TF-IDF value in order to propose new concepts for the extension of an ontology, which also modifies this ontology. Still, ontology learning shows some interesting techniques for information extraction (clustering, TF-IDF weighting). Khadir et al. [24] list some recent machine learning and deep-learning approaches that are useful to extract relations and axioms adequate to feed an ontology, even though the final goal is still ontology learning and not ontology population.

Learning concepts and relations The progress in the depiction of dependencies between terms within unstructured text has been highly supported by machine learning [25]. Syntactic parsing can subsequently be used as a medium for relation extraction [1,3,22]. Lomov et al. [30] trained a neural network to recognize concepts that are similar to concepts already existing in an ontology. Among the detected concepts, some are then derived as instances of the ontology, allowing semi-automatic ontology population. Nguyen et al. [32] used an objective function traducing how often a term appears within a given context in order to build a statistical representation of hypernyms. From this representation, statistical analysis and support vector machines training can then be applied to classify whether two terms are hypernyms or not.

Using embeddings as an intermediate The past few years have witnessed the semantic enrichment of textual data, notably through the appearance of word embeddings. Word2Vec, GloVe or BERT are all deep-learning based embedding models that present the possibility to express the semantic dimension of terms, each at different levels of precision. With the emergence of neural network using attention mechanism [40], semantic representations can be learnt and used to compute semantic meaning for bigger sequences of textual data [42] or for the detection of relations between terms [16,31]. Zhang et al. [43] used several neural networks to learn protein-drug interactions from word vectors. OntoEnricher [36] uses the Universal Sentence Encoder resulting embedding [4] to train a feed-forward neural network to learn different kinds of relations such as hypernymy and hyponymy relation from a Wikipedia corpus. The training vectors are built based on a combination of both dependency paths and word embeddings and used to proceed a classification of the corresponding relation. On2Vec [8] directly gives a representation of relations using existing knowledge bases. By opposition with other translation-based models, On2Vec takes into account the different characteristics of a relation, such as symmetry or transitivity by redefining the energy function of the model. All the presented systems are either supported by a pre-existing knowledge base, or trained on annotated datasets resulting in supervised models. However, the main limit of supervised techniques is the need of annotated data, that are not available for every domain or even for every relation/concept of a domain.

Need of a dedicated performance measure The fact that most of the techniques are designed for supervised applications induces that they are evaluated on gold standard datasets. Beyond the limitations due to the restricted areas in which these datasets are available making them not suitable for every domain of interest, some limitations are linked to the fact that these datasets can not provide a sufficiently good evaluation of the performance of a system using unsupervised techniques for relation detection. Nevertheless, these gold standard resources remain interesting as they contain knowledge that can be used for system evaluation. This is the main motivation that conducted to find a new evaluation strategy in order to estimate the performance of an unsupervised system from existing sources of knowledge.

2.2. Proposed methods for extraction and performance evaluation

Two constraints can be derived from the limits identified in the previously presented studies. The first one concerns the need of genericity regarding the domain and the source of explored data, and the second one is the need to avoid supervised methods. Some approaches satisfy both genericity and unsupervised context constraints. However, as it is shown in Fig. 1, these approaches are not necessarily guided by an ontology. Some of them even build an ontology from scratch, losing the initial expertise that is contained in an ontology.

As a result, this paper introduces an extraction system that takes advantage of existing concepts of an ontology and uses adapted Hearst patterns to generically detect hypernym relations. However, respecting the constraint of unsupervised learning implies other difficulties concerning the performance evaluation of such an extraction system as reference data are necessarily different from extracted data. To overcome these limits this paper also proposes a performance evaluation method compliant with unsupervised context.

3. A generic framework for ontology population

This section is dedicated to the description of a framework for ontology population. This framework contains an unsupervised rule-based relation extraction system, which constitutes the first contribution of this paper. The need to validate the extracted couples also conducted to a second contribution which consists in the automated validation of those extracted couples. However, this validation method is not included in the framework on purpose, as it can be applied to any set of extracted relations and not only to a group of couples extracted with the presented system.

3.1. Overview of the framework

In order to perform unsupervised information extraction independently of the data source and the domain to which the targeted ontology refers, a methodological framework has been defined, based on the general principles of model-driven engineering. This methodological framework is built upon a generic metamodel for the representation of heterogeneous data. This metamodel, presented in Chasseray et al. [6], allocates the extracted data within six classes: *Ontological Object*, *Concept*, *Instance*, *Relation*, *Extracted Data* and *Context*. In this metamodel, the *Concept* class is used to retrieve the elements representing the classes of the ontology to be populated from the analyzed data. The *Instance* class allows to retrieve the instances related to these ontology classes (derived as *Concepts*) from the data, which will eventually become individuals of the ontology. The *Concept* and *Instance* classes, both inherited from the *Ontological Object* class, are linked by the *Relation* class which qualifies the nature of the relation identified between two occurrences of the *Ontological Object* class. Meanwhile, the *Extracted data* and *Context* classes allow to embed additional information on the extracted concepts and instances. This enables to capture, on the one hand, the raw data from which the element is extracted, and on the other hand, contextual information that semantically enrich the extracted element.

These six classes thus provide enough information to carry out transformations towards a target ontology. The general framework in which this metamodel is integrated is presented in Fig. 2. This framework is composed of two processing pipelines which, although they can work independently of each other, are complementary as they allow the combination of both semantic and rule-based approaches. This framework contains iterative loops, the process being initiated by the rule-based approach which is then joined and completed by the semantic approach. The two retroactive loops are fed by the instances extracted during the initial step in order to, on the one side, automatically deduce new extraction rules, and on the other side, derive semantic matchings. Further details about this framework are given in [6].

Section 4 of this paper focuses on the rule-based retroactive loop. Thus, the initial extraction process and the process of deducing new extraction rules, using respectively the extraction schemes proposed by Hearst [20] and the bootstrapping principle introduced by Pennacchiotti and Pantel [34] are presented in section 4.

3.2. Framework specification for the processing of textual data

The generic framework presented in section 3.1 can be specified for textual data processing through the use of automatic language processing techniques. In this section, we briefly present the automatic language processing methods that are used to deduce a data model from the metamodel and textual data and to feed the framework retroactive loops. Thus, in Fig. 3, a standard processing pipeline that subsequently gives place to three specific processing pipelines is presented. The standard processing pipeline contains classical tokenization and tagging operations. Based on this pre-processing step, the differentiated processing pipelines (1), (2), and (3) are applied with their own purpose. The processing pipeline labelled (1) in Fig. 3 is designed to extract pairs of entities linked by a relation. This relation can be of different natures (hyponymy, hyperonymy, other domain specific relations defined by the ontology). These couples will then be used to instantiate the data model and finally be attached to the ontology. This processing pipeline contains three steps. The first one allows to identify and label concepts that appear in the text. This identification is driven by the concepts extracted from the ontology. The second step consists in building a dependency tree which represents the syntactic dependencies that exist between the tokens of the text. Based on the identified concepts and this dependency tree, extraction schemes are applied in the third processing step. This processing pipeline takes place in the system initialization step mentioned in the previous sections.

The extraction pipeline labelled (2) in Fig. 3 allows to extract other entities, this time not linked to each other. The extraction is done according to two filters. The first filter focuses on the Part Of Speech-tags (POS-tags) representing terms that can be related to

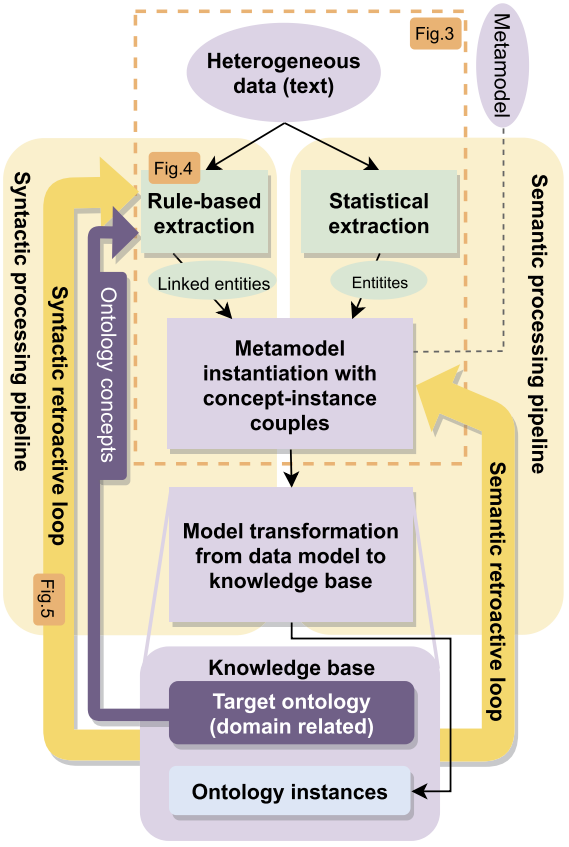

Fig. 2. Generic framework for domain independent ontology population.

instances, which are essentially nouns. A second filter, statistical, selects the entities that are susceptible to be instances related to concepts of the domain, as defined by the ontology. Although the search concerns instances of the domain, this extraction pipeline is not guided by the ontology. Thus, the resulting entities will not be directly linked to the ontology but will have to undergo a semantic matching step and will therefore feed the semantic retroactive loop which is in charge of this semantic matching.

Finally, the third specific processing pipeline, labelled (3) in Fig. 3, has a slightly different objective from the first two, since it is intended to extract all the context elements (co-occurrences, semantic vectors) that make it possible to characterize the entities previously extracted. To achieve this, the processing is done in three steps. The first step consists in eliminating the stop-words in order to focus on the meaningful terms. The second step consists in the building of the co-occurrence matrix that can be used to link terms that co-occur with an extracted entity or an ontology concept for example. Third step uses a pretrained language model [11] to obtain a vector representation of the extracted entities. These context elements are intended to be reused later to identify instances among the unrelated entities in the semantic retroactive loop. In this paper, it has been chosen to detail the extraction and deduction methods used for the operation of the rule-based retroactive loop. This loop is based on the processing pipeline labelled (1). An application of syntactic pattern extraction on textual data is presented in section 4.1 which details the way to proceed an extraction from concepts extracted from an ontology.

4. Pattern extraction from ontology concepts

In many domains, there is little or no annotated data already available that would allow a supervised approach to knowledge extraction. Thus, a generic ontology population system must be able to operate in an unsupervised context, i.e. without previous knowledge about the entities to be extracted. Furthermore, this system must be adaptable to different domains. The rule-based approach with the use of generic extraction schemes then gains its full meaning. In this section, the definition of generic extraction patterns for textual data extraction is explained and the operation of the rule-based retroactive loop including the deduction of new patterns is illustrated. The proposed extraction method is compared to the state-of-the-art supervised BERT-Tagger model [12], which consists of a linear classifier based on BERT model's outputs [11]. Comparisons are made in terms of performance and user experience.

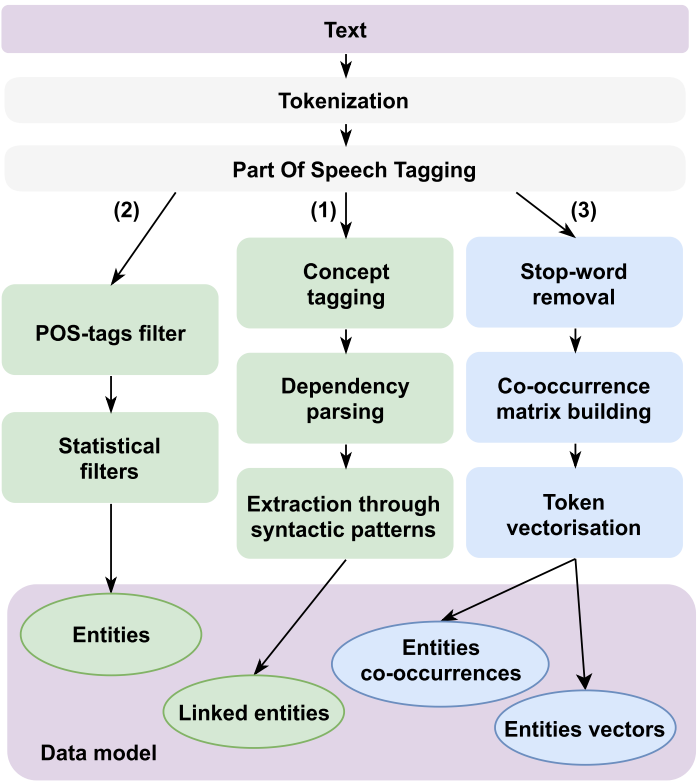

Fig. 3. Natural Language Processing pipeline processed to build a data model from textual data.

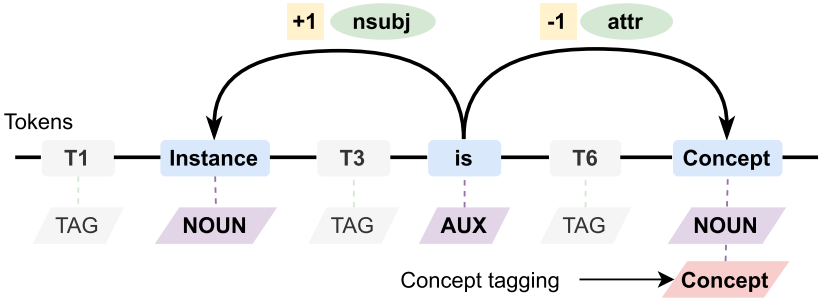

Fig. 4. Example of a generic extraction pattern for hyponymy relation extraction.

4.1. Initialisation with generic extraction patterns

As discussed in section 3, the rule-based feedback loop – and the overall system by extension – contains an initialization step for extracting the first instances. This step is crucial because it conditions the future learning of new extraction patterns and must also guarantee the performance of the semantic retroactive loop. Fig. 4 illustrates a syntactic extraction pattern based on Hearst's [20] patterns which represents the hyponymy relation that can appear in the text between a concept and its instance. These extraction patterns are actually built as the superposition of three extraction sequences each one acting like a filter.

The first sequence, named p is related to the POS-tags and describes the sequence of tags requested. In the example shown in Fig. 4, the targeted sequence of POS-tags is the following: $(Concept) \rightarrow (AUX) \rightarrow ((NOUN) \vee (PROPN) \vee (PROP))$.

The second sequence, named d relates to the tags assigned to the dependencies in the syntactic dependency tree and therefore describes the sequence of syntactic dependencies searched. In this example, the sequence of dependencies searched is the following: $(attr) \rightarrow (nsubj)$.

A third sequence, named s allows to specify the direction of navigation in the syntactic dependency tree, i.e. to indicate whether the pattern goes up (-1) or down $(+1)$ the dependency tree. It is important to specify that the order of the extraction scheme does not necessarily coincide with the reading order of the studied text. In the proposed example, the sequence used to specify the navigation is the following: $(-1) \rightarrow (+1)$.

Beyond the use of syntactic dependencies to bring more genericity, this way of defining extraction patterns contains a particularity. Indeed, if the sequence of tags used is mainly made of classical POS-tags, an additional tag is used to indicate that the token represents a concept of the ontology. This is especially this tag that triggers the search for a relation involving this concept following the predefined generic pattern. The process of labeling concepts based on of the ontology's classes to ensure the identification of these concepts is performed upstream, in the automatic language processing pipeline (see Fig. 3).

If each of the three layers of the pattern can be respected by the preprocessed textual data, then one – or more depending on the pattern – instances can be detected. In the given example, a concept and an instance will be systematically detected at both ends of the extraction pattern, resulting in the creation of two ontological objects linked by a hyponymy relationship in the data model.

The generic and standard definition of patterns is useful as it allows to apply any pattern following this definition with the same automated algorithm. The Algorithm 1 is a generic algorithm that can be used to apply a pattern, as previously formally defined, once a concept (*tok*) has been found. This algorithm recursively checks if parent or children of explored token fit in the given pattern until the last group of tokens of the sequence is found and becomes an instance. The Structure 1 defines a token, that is used as the main structure in the Algorithm 1. This definition follows the technical definition of a token, the *head* attribute being the parent of a token in the dependency parsing tree, the *dep* list attribute being the list of dependency tags linking the token to its children and the *pos* attribute being the POS-tag of the token.

Structure 1: Token structure.

```
Struct Token contains
    Token head;
    list of char dep;
    char pos;
```

Algorithm 1: ApplyPatterns (AP).

```
Data: d: dependency sequence
      p: part-of-speech sequence
      n: navigation sequence
      tok: current token (starting with concept)
      idx: integer indicating the position in the pattern
      children: list of token, children from the current token
Result: instances : list of detected instances
begin
    instances ⟵ 0
    idx ⟵ idx + 1
    if idx = length(s) then
        instances ⟵ getInstanceText(tok)
    else
        if s[idx] = 1 then
            children ⟵ getChildrenByDep(tok, d[idx])
            for child ∈ children do
                if child.pos is in p[idx] then
                    subI ⟵ AP(s, d, p, child, idx)
                    extend(instances, subI)
        else
            if tok.dep is in d[idx] and tok.head.pos is in p[idx] then
                subI ⟵ AP(s, d, p, tok.head, idx)
                extend(instances, subI)
```

4.1.1. Applied patterns

In the implemented version of the framework, three patterns for hyponymy relation detection have been defined based on Hearst's [20] patterns.

- *I aux C*: This pattern allows to capture entities that are explicitly described by a hyponymy relation in the text. These relations can be expressed through any auxiliary (be, should, would, must). This pattern can detect relations expressed as follows: *I is a C*, *I must be a C* where I is a NOUN, and C is a NOUN that has previously been identified as a concept. Following the formal definition of a pattern given in section 4, the *I aux C* pattern is expressed with the three following sequences:
  - ✦ $p = [\,[\,aux\,],[\,prop,\ propn,\ noun\,]\,]$
  - ✦ $d = [\,[\,attr\,],[\,nsubj\,]\,]$
  - ✦ $s = [-1,\ +1]$

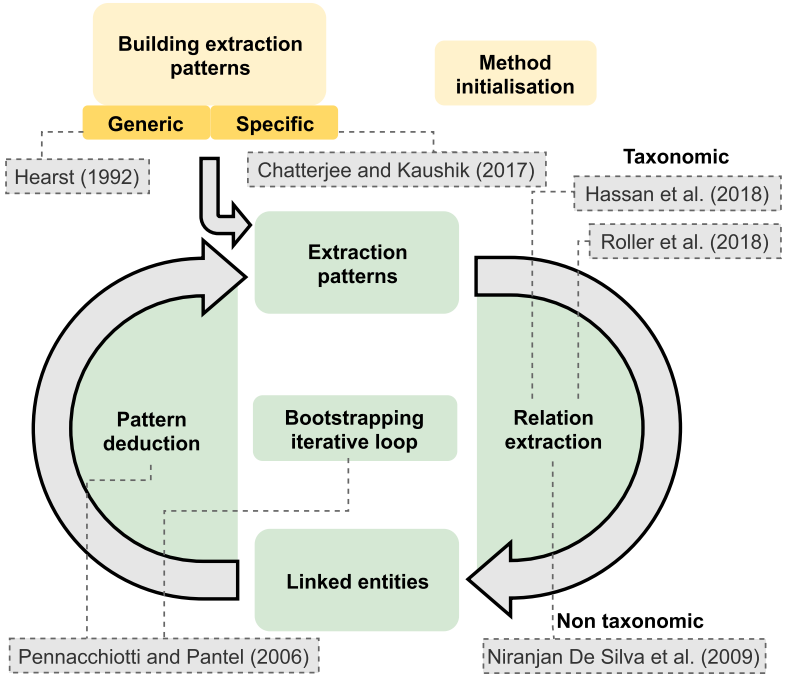

Fig. 5. Illustration of bootstrapping approach for pattern deduction and rule-based extraction.

- *C prep I*: This pattern detects instances taken as an example for a given concept. One of the specificity of this pattern, is its ability of detecting a sequence of several instances once the first has been found. As it has been said already, using syntactic tags instead of terms in the pattern allows to broaden the range of matching sequences. In this case, the same pattern is able to match both expressions of the form *C 'such as'* I_1 *'or'* I_2, and *C 'like'* I_1, I_2. The *C prep I* pattern is defined through the three following sequences:
  - $p = [\,[\,sconj,\ adp\,],[\,prop,\ propn,\ noun\,]\,]$
  - $d = [\,[\,prep\,],[\,pobj\,]\,]$
  - $s = [+1,\ +1]$
- I = *modifier* + C: This third pattern is used to detect instances as they appear as an extension of a given concept. This pattern is triggered as soon as a concept has a modifier directly linked to it that expresses a specification of this concept. The grammatical nature of the modifier is not restricted in the definition of the pattern as it can be either a name, an adverb, a verb or an adjective. Then the pattern I = *modifier*+ C can be generically expressed through the three following sequences:
  - $p = [\,[\,amod,\ compound,\ npadvmod,\ advmod\,]\,]$
  - $d = [\,[\,any\,]\,]$
  - $s = [+1]$

These three patterns have been implemented in the global framework. In association with the implemented version of the Algorithm 1, these patterns can then directly be applied to textual data.

4.2. Extraction pattern deduction mechanism

Generic and specific rule-based extraction is a method that has been investigated for a few years. Fig. 5 gives a global illustration of how the rule-based retroactive loop works within the extraction framework. This figure presents the bootstrapping principle proposed by Pennacchiotti and Pantel [34].

This method consists in a mutual feeding of the rule deduction and rule application steps. The deduction of new rules is done by inverting the extraction process. When previously identified instances appear linked to their concept in the data, the bootstrapping principle consists in deducing a pattern that links these two instances. If the deduced pattern is representative of the relationship expressed between two instances, or between an instance and its associated concept, then it can be added to the set of patterns. It can then be used to detect new relations in the data. This method therefore assumes the existence of identified concept-instance pairs, underlying the importance of the initialization step and the use of generic human-defined patterns.

In the context of textual data processing, the rule presented in this article is a syntactic pattern, applied to the dependency tree previously built. However, this approach can be extended to other types of rules, that are applied to other types of data (structure-based rule, tag chaining). In the case of syntactic patterns, additional filters have also been added to the bootstrapping method in order to avoid the discovery of dependency path that would be too long to be indicative of an hyponymy relation. The

Table 1
Comparison of the proposed system and BERT-Tagger on Few-NERD dataset.

	Proposed extraction system	BERT-Tagger [12]
Precision	71.29	65.56
Recall	4.28	68.78
Labelling step	No	Yes
Model training step	No	Yes
Evaluation method	Manual	Label-based
Approach	Unsupervised	Supervised

maximum accepted length of a dependency path has been set to 5 dependencies by the authors as a compromise between relevance of the rules and reduction of computing time.

4.3. Comparison with BERT-Tagger model

In order to illustrate the differences between our extraction system and other supervised methods, we applied our extraction patterns on a dataset and proceeded to a manual evaluation of extracted instances. Obtained results are compared with the state-of-the-art BERT-Tagger, and put into perspective regarding the nature of each method.

4.3.1. Dataset description

The chosen dataset is the Few-NERD dataset for supervised learning. This dataset meets our needs for two reasons:

- It is annotated with 66 fine-grained labels. This allows the transcription into an ontology with enough classes to effectively apply our extraction method.
- It has already been used in the literature for supervised learning performance measurement and has published results [12].

The 66 fine-grained labels are grouped into 8 coarser labels: (1) Location (example: Island, Moutain), (2) Organization (example: Company, Media), (3) Event (example: Disaster, Election), (4) Building (example: Airport, Hospital), (5) Art (example: Film, Music), (6) Product (example: Car, Food), (7) Person (Actor, Athlete) and (8) Miscellaneous (example: Disease, Language).

4.3.2. Evaluation methods

As the used dataset has initially been built for model training on named entity recognition task, the extraction results can be compared with our proposal. Because the proposed method is unsupervised and does not use the datasets' labels, manual validation is required. Hence, to keep this step achievable in a reasonable amount of time, a sample of the original dataset has been selected. This sample contains a thousand evaluation sentences. During the evaluation process, each extracted relation is examined within its context to be either accepted or rejected. Precision is then measured by rating the number of accepted relations over the number of extracted relations. Recall value is computed by rating the number of accepted relations over the number of previously annotated relations in the evaluation dataset.

4.3.3. Comparison results

The obtained precision and rate of extracted instances are compared to published results. Table 1 compares performances and characteristics of each method. Ding et al. [12] trained the BERT-Tagger model on this dataset and obtained a precision of 65.56 and a recall of 68.78. Our system performs extraction with a reasonable precision score. However it is not able to extract as much knowledge as a system – like BERT-Tagger – that has specifically been trained on the labelled dataset can. This can be explained by the fact that the proposed rule-based extraction method supposes the appearance of the ontology class in the sentence to detect a paired instance, which occurs only in a few sentences of the dataset.

Nevertheless, the proposed system has been designed to be, at the same time, unsupervised and not related to a specific domain. The interest of such a system does not lie in its unsupervised or generic nature only, but in the combination of both characteristics. Taking these two aspects into account, the provided methodology allows to extract relations from any technical textual data without previous model training or data labelling task, as reminded in Table 1. The hidden objective is to provide access to extraction tools for users who already have a knowledge structure and need to explore their data within a limited amount of time. Then, the generic nature of the approach allows to apply the extraction method quickly on any knowledge structure, without regarding the described domain. On the contrary, supervised models will be limited to the domain and ontology classes they have been trained on or will always require additional annotated data to achieve good performance on a new domain.

5. Performance measure in an unsupervised context

The lack of annotated data as an obstacle to the adoption of a supervised approach has already been mentioned in the introduction of section 4. This problem similarly affects the approaches that concern the evaluation of an extraction system. Indeed, in automated data processing tasks, the performance of a method is generally evaluated by applying it to a reference dataset (commonly identified as a test set). Results obtained by the automated process are then compared to ground truth. The computing of a distance between

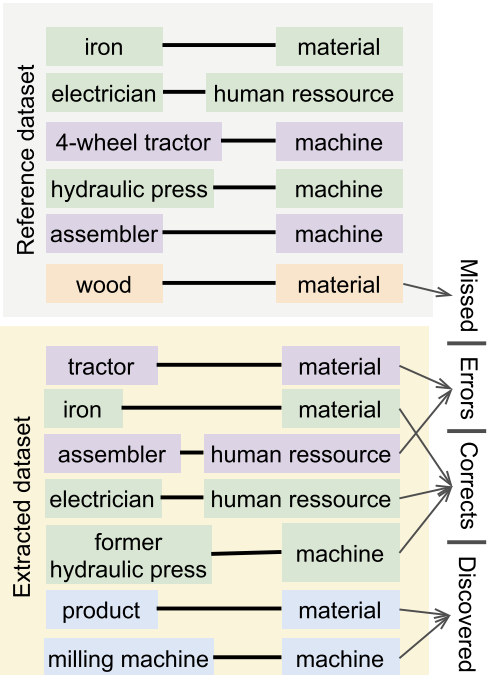

Fig. 6. Representation of the applicative example sample and associated sets.

these two results provides an evaluation of the performance of the system. This is the case, for example, in many machine learning tasks that use evaluation methods such as the calculation of the F1-Score and the drawing of ROC curves [13]. Unfortunately, in a context where annotated data is not widespread, and with unsupervised methods such as the one presented in this paper, these methods are limited due to several reasons:

- They do not allow to evaluate the performance of a system when it is applied to new data sets.
- They do not take into account new knowledge, possibly extracted by the system but missing (or ignored) during the annotation.
- They are biased by the way data have been annotated and are very sensitive to the subjective dimension of expert annotation.

Hence, the second contribution of this article is to deal with these aspects by defining a performance measure that does not strictly apply to extractions made on a given and predefined annotated test set.

One will therefore assume that the data that are processed are not – or very rarely – annotated. Two solutions can then be considered. The first one is to manually label the data once the extraction has been completed. This method requires a strong investment of domain experts specialized in the concerned field, but allows in return to determine the accuracy of the system in a rigorous way. The second method relies on existing reference data independently of the population process. This second method has the advantage of not requiring the domain expert to perform a manual validation. However, while it can be implemented rapidly, it does not allow a rigorous determination of the accuracy of the model. This section focuses on the second method, which is applied in the context of concept-instance relations and which requires a redefinition of the performance measures. To address this problem, it is necessary to clarify the notions of *concept*, *instance* and *couple*.

A *concept* is here the representation in the data (reference or extracted) of one of the classes initially contained in the ontology. An *instance* (related to a concept) is the representation in the data of the instance of a class of the ontology, which has been extracted or which is part of the reference data. Finally, a *couple* is the union of a concept and an instance (either from the extraction or from the reference data) linked by a relation. To illustrate the measure, it will be applied to the sets of simulated extracted and reference couples presented in Fig. 6. It is important to specify that these sets of couples are used as examples, and are not linked to any particular dataset. Nevertheless, performance measurements using this method have been performed on real datasets. The corresponding results are presented and discussed in section 5.5 and 5.6.

5.1. Reconciliation of extracted and reference data by the ROUGE Score

The evaluation of the system must therefore be possible from reference data that are not necessarily related to the studied text, but that constitute a correct representation of the knowledge base that is targeted through the population of the targeted domain ontology. The problem induced by this context is that the non exact matching of the extracted data with the reference data does not mean that the extraction system made a mistake. Indeed, since characters sequences (representative of the extracted instances) are compared, it can happen that some of them slightly differ without changing their meaning. In order to avoid misjudgments due

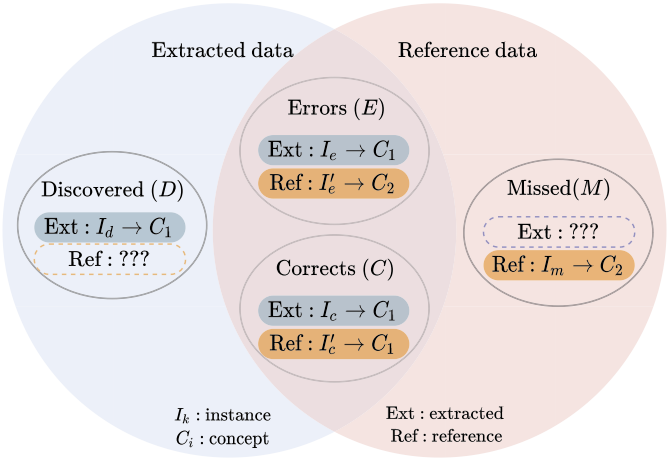

Fig. 7. Illustration of C, E, M and D sets definition.

to this specificity, the adopted solution is to define a distance between the strings and to apply a threshold determining when two strings are designating the same entity.

Many metrics exist and can be used to compute the distance between two strings. Among them, Levenshtein's distance [27] or Hamming's distance [18] are the references. Boufrida and Boufaida [2] use the Jaro-Wrinkler distance [41] in order to support an algorithm that matches terms of a text to ontological entities. However, these distances are generally used on strings that are either already very similar or very long, which is not necessarily the case when comparing extracted and reference instances. Moreover, these measures can be very inefficient for the comparison of strings that are close from a purely character-based point of view but have completely different semantic meaning. For example, the strings *iron* and *wood* have a *relatively* small Levenshtein's distance (Levenshtein(*wood*, *iron*) = 6)) but represent different instances.

ROUGE Score [28] between two strings was therefore chosen to characterize the equality of the instances they represent. ROUGE Score is a simple similarity index that adopts a measurement method similar to Levenshtein's distance except that the latter is adapted to the term level (and not to the character level). In this manner, the strings *iron* and *wood* appear to be farther apart (ROUGE Score = 0) than the strings *hydraulic press* and *former hydraulic press*, for example (ROUGE Score = 0.79), which is not the case when the distance is computed at the character level as in the case of previously mentioned distance measures (Levenshtein(*wood*, *iron*) = 6, Levenshtein(*hydraulic press*, *former hydraulic press*) = 7). Coupled with an acceptance threshold, the ROUGE Score value then acts as an indicator of the equality between two instances represented by two strings of characters.

5.2. Building Correct, Missed, Discovered and Errors sets

Once the method of comparison between the extracted data and the reference data is established, it is possible to categorize the extracted instances according to whether or not they have been assigned to the matching concept in the reference data. As the reference data are not necessarily related to the text used for the extraction, it can happen that some instances of the reference data are not found in the extracted data and vice versa, that some extracted data are not represented in the reference data. Thus, four sets are defined in this section to reflect this specificity:

- ***Corrects*** (C): The *Corrects* set contains the couples whose instances are considered identical in the reference dataset and in the extracted dataset and for which the associated concept is consistent with the concept of the reference set.
- ***Errors*** (E): The *Errors* set contains the extracted couples whose instances appear in the annotated set but is associated with a concept different from the one mentioned in the annotated set, either by error or by over-classification.
- ***Missed*** (M): The *Missed* set contains the couples in the reference dataset that do not appear in the extracted dataset, because the instance was not detected at all in the extracted data.
- ***Discovered*** (D): The *Discovered* set contains all the couples identified by the system but whose instances do not appear in the annotated dataset.

Fig. 7 provides an illustration of these sets. On this representation, it appears clearly that sets M and D are made of couples that do not have a corresponding couple in extracted (for M) or reference (for D) dataset.

The equality between two instances is stated when the distance between these two instances passes under a threshold defined by the evaluator. The distance used is based on the ROUGE Score. In the given example, the threshold value is set to 0.6. Sets E and C contain extracted couples whose instances are equal in the sense of the defined distance. The set D contains extracted couples whose instances are not similar enough to any of the reference instances to be classified in one of the sets C or E. The set M, on the other hand, contains reference couples that do not appear in the extracted data and cannot be classified in either C, E or D.

Table 2
Automated evaluation results for the example datasets.

	Human resource	Machine	Material
TP	1	0.79	1
FP	1	1	1.66
TN	5.45	5.66	3.79
FN	0	0	1
Precision	0.50	0.56	0.38
Recall	1	1.00	0.5
F1-Score	0.67	0.72	0.43

The choice of not including in M the couples whose instances appear – in the extracted couples – associated with a different concept is to distinguish the missed couples from the incorrectly defined ones (error made on the concept matching). This decision induces a bias because it does not take into account the particular cases where an instance has been linked by the system to only a part of the concepts present in the couples involving the instance in the reference data. For the sake of clarity, this case is not presented in the example.

5.3. Precision and recall estimation

The sets M and D presented in section 5.2 are singular because they involve couples whose instances are present in only one of the two datasets. Couples can thus appear in the M set either because the instance associated with the concept does not appear in the data and could therefore not be extracted, or because it was not extracted, despite its presence in the text. In the first case, the couple cannot be considered as a real failure of the system. In the second case, it is indeed an instance that has not been extracted.

Similarly, it is difficult to estimate the value of the couples contained in the set D which may be either new relevant couples, which were not listed in the reference data, or a set of errors from the system (false positives).

In the absence of manual annotation, it then becomes difficult to establish the exact performance of the system without making assumptions about the nature of the couples contained in sets M and D. It is however possible to give an estimation by assuming that the couples in set M are missed couples and that the couples in set D, as they are not part of the reference set, do not constitute sufficient knowledge to be considered as true positives. With these assumptions, Ens_{c_i} and $\overline{Ens_{c_i}}$ subsets can generically be defined from previously presented sets as follows:

$$Ens_{c_i} = \{(con, ins) \in Ens \mid con = c_i\} \tag{1}$$

$$\overline{Ens_{c_i}} = \{(con, ins) \in Ens \mid con \neq c_i\} \tag{2}$$

Confusion matrices can then be defined from these subsets. For each concept (C_i), true positives are estimated from C_{c_i} subset, false positives are estimated from E_{c_i} and D_{c_i} subsets, true negatives from $\overline{E_{c_i}}$, $\overline{D_{c_i}}$ and $\overline{C_{c_i}}$ subsets and false negatives from M_{c_i} subset:

$$TP_{c_i} = \sum_{cpl \in C_{c_i}} sim_{cpl} \tag{3}$$

$$FP_{c_i} = \sum_{cpl \in E_{c_i} \cup D_{c_i}} sim_{cpl} \tag{4}$$

$$FN_{c_i} = \sum_{cpl \in M_{c_i}} sim_{cpl} \tag{5}$$

$$TN_{c_i} = \sum_{cpl \in \overline{C_{c_i}} \cup \overline{E_{c_i}} \cup \overline{D_{c_i}} \cup \overline{M_{c_i}}} sim_{cpl} \tag{6}$$

Where sim_{cpl} is valued 1 for couples in sets D and M and is the similarity index computed from the ROUGE Score between an extracted couple and its counterpart within the reference data for couples in sets C and E.

Indeed, for the couples belonging to the sets C and E, the equality is stated from a similarity value. To take into account this similarity, rather than adding a unit to the groups of false positives or true positives, the ROUGE Score value is used. This measure, between 0.6 (threshold value) and 1, is an indicator of the degree of similarity between the two couples. Thus, a couple considered as belonging to the set C because it has just passed the admission threshold will have less weight than a couple for which the ROUGE Score value is 1 because the latter is perfectly identical to the reference couple. In the presented example, the values detailed in Table 2 for each of the concepts presented in Fig. 6 are obtained using this weighting method.

5.4. Running the example

An example of the sets described before is given for *material* concept (abbreviated *mat*) in Fig. 8. For C_{mat}, E_{mat}, $\overline{C_{mat}}$ and $\overline{E_{mat}}$, a similarity value – computed with the ROUGE Score – is associated to each couple in order to express their weight in

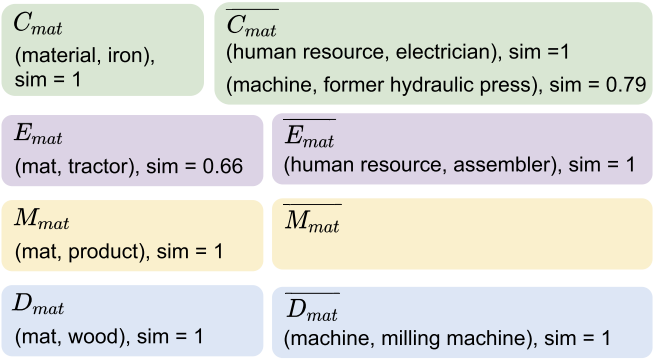

Fig. 8. Attribution to couples and associated similarity values for the concept *material*.

each set.[3] As explained before, the ROUGE Score is computed between the instance of the given couple and the instance of the corresponding couple appearing in the reference dataset. For example, for couple (*material, tractor*), the corresponding couple is (*machine, 4-wheel tractor*) which means that $Sim_{(machine,human\,resource)}$ is obtained by comparing *tractor* and *4-wheel tractor instances* ($ROUGE(tractor, 4-wheel\,tractor) = 0.66$).

In the case of the concept *material*, TP, FP, TN and FN values can be computed as presented in equations (7) to (10). From there, *Precision*, *Recall* and *F1-Score* can be computed as well (equations (11) to (13)).

$$TP_{mat} = Sim_{(mat,iron)} = 1 \tag{7}$$

$$FN_{mat} = Sim_{(mat,product)} = 1 \tag{8}$$

$$FP_{mat} = Sim_{(mat,tractor)} + Sim_{(mat,wood)} = 1.66 \tag{9}$$

$$\begin{aligned} TN_{mat} =& Sim_{(machine,former\,hydraulic\,press)} + Sim_{(human\,resource,electrician)} \\ &+ Sim_{(human\,resource,assembler)} + Sim_{(machine,milling\,machine)} = 3.79 \end{aligned} \tag{10}$$

$$P_{mat} = \frac{TP_{mat}}{TP_{mat} + FP_{mat}} = 0.38 \tag{11}$$

$$R_{mat} = \frac{TP_{mat}}{TP_{mat} + FN_{mat}} = 0.5 \tag{12}$$

$$F1_{mat} = \frac{2 * P_{mat} * R_{mat}}{P_{mat} + R_{mat}} = 0.43 \tag{13}$$

After proceeding for each concept, the global F1-Score can be computed by weighting the F1-Scores of each concept with the participation (i.e. similarity value) of each of the couples in which the concept intervenes. Details of this weighting are given in equation (14). In this simple example, the obtained global F1-Score is equal to 0.56 (see Table 2).

$$F1 = \underbrace{0.67}_{F1_{human\,resource}} * \overbrace{0.27}^{w_{human\,resource}} + \underbrace{0.72}_{F1_{machine}} * \overbrace{0.24}^{w_{machine}} + \underbrace{0.43}_{F1_{material}} * \overbrace{0.49}^{w_{material}} = 0.56 \tag{14}$$

5.5. Application to biochemistry datasets

The method illustrated in the previous sections has been tested using a reference dataset related to the field of biochemistry (200 annotated abstracts) and performing extraction on research articles associated with this reference dataset (100 annotated articles) [38]. It is important to note that the reference data and the data on which the extraction is applied are distinct, highlighting the interest of the proposed method.

Table 3 shows the volume (sum of similarities) of sets C, E, M and D for a threshold of equality set at 0.5. We obtain for this data set a precision value of 0.65. However, due to the method and for the reasons mentioned in section 5.3, this is a pessimistic evaluation of the precision. The recall value is very low (about 5%) since many couples are considered as missed by the extraction. Again, this strict estimation is discussed in section 5.3.

Despite the extraction data contains more text than the reference abstracts, the reference annotated couples are way more numerous (> 5000 couples) than the extracted couples (307 couples). This is mainly due to the nature of the used extraction techniques which are restricted to few generic patterns, whereas manual annotation allows to extract all relevant elements from

[3] For other sets, this similarity value is equal to 1 for every couple.

Table 3
Summary of set repartition of couples after automated evaluation for biochemistry datasets.

	Volume	Rate
Corrects (C)	142.27	5.2%
Errors (E)	10.83	0.4%
Missed (M)	2503	91.5%
Discovered (D)	78	2.9%

Table 4
Summary of set repartition of couples after automated evaluation for Fukushima crisis datasets.

	Volume	Rate
Corrects (C)	13.07	3.49%
Errors (E)	2.77	0.74%
Missed (M)	38	10.14%
Discovered (D)	321	85.64%

the text. As a result of this imbalance, there is only a little part of the overall data that has been affected to sets C and E. This number could be increased by reducing the threshold parameter, but the risk would be to wrongly affect some couples to C or E sets. Nevertheless, it remains possible to conclude that the evaluation method can be applied to two datasets of couples that have been built from different documents.

All reference couples that couldn't be paired with an extracted couple, either because it has not been extracted or because it did not appear in the data used for extraction end up in the missed set which, considering the restrictive aspect of the rules, results in an important rate of missed couples. Another explanation to this important amount of missed couples is that, in the case of annotated data, many couples are annotated each time they are encountered in the text. A term which never appears in the extraction data is consequently considered as missed each time it has been annotated.

When looking manually closer at the type of couples that are extracted by the system and sorted in the different sets, further analysis can be made and some specificity of the evaluation system can be underlined considering C and E sets. In the reference dataset, identified concepts are *Chemical*, *Metabolite*, *Spectral data*, *Protein*, *Specie* and *Biological activity*. The ranges of these concepts are not exclusive, meaning that the same instance may belong to several concepts at once in the reference dataset depending on the expert's proper definition of each concept. The risk implied by this specificity is for a couple of the extracted dataset to be identified as errors when they are actually associated to only one concept. In order to handle this issue, it has been decided to consider that an instance associated to one concept in the extracted dataset and two (or more) concepts in the reference dataset can be affected to the *Correct* set if at least one of the concepts corresponds. The other couples are however considered as missing in the extracted data. This rule, is well integrated in the evaluation method as it is defined in section 5.2.

Still, another problematic specificity is encountered when a concept's definition includes other concepts, which is the case in this application, where the term *metabolite* has been annotated in the reference dataset as an instance of the concept *Chemical*. This particularity – associated with the fact that couple's affiliation is decided over similarity – leads to evaluation mistakes. For example, the instance *sole metabolite* from the couple *sole metabolite – metabolite* is (unfortunately) matched with the instance *metabolite* from the couple *metabolite-chemical* as no closer instance exists in the reference dataset. Despite the extracted couple seems to be correct, the fact that it is paired with a couple in which the concept doesn't match leads to its classification as an error.

It is a choice of the authors not to alter the reference dataset during the evaluation as (i) specific knowledge is often required and (2) it would imply a specific treatment of the reference dataset whereas the first objective of the method is to remain generic and compatible with any reference dataset. However, this kind of issue should be taking into account when processing automatic evaluation, by avoiding couples that define concepts as instances of another concept.

5.6. Application to crisis management datasets

To apply the evaluation method on crisis management domain, we used data extracted from two different sources. Reference dataset is built on validated concept-instance relations initially extracted from online press articles from the New York Times. The reference corpus is made of 20 web pages from the New York Times website containing one article each. The extraction and validation from these press articles resulted in 59 reference couples. Then extracted dataset is built with data extracted from seven scientific articles about Fukushima crisis resulting in 341 couples. The extracted relations have not yet been validated in order to proceed validation from reference couples. Validated relations from scientific articles are thus used as a reference to seek which elements of the extracted data can be automatically affected to set C or E. Table 4 shows the volume and relative importance of each set deduced from the comparison between extracted data and validated data with a ROUGE-Score threshold value fixed to 0.5. The relatively low volume of E and C sets can be explained by the fact that the two sources of data differ and that reference data do not cover all the relation of the domain. Still, Table 5 shows the couples of sets C and E that have been paired through the evaluation of the ROUGE Score. Some paired couples show the limitation of this metric and offer elements to be discussed.

Table 5
Details about paired couples of sets C and E.

Set	Extracted (scientific articles)		Reference (press articles)		ROUGE
	Instance	Concept	Instance	Concept	
Corrects	high radiation area	area	high - risk area	area	0.57
Corrects	health risk	risk	health risk	risk	1.00
Corrects	nuclear plant accident	accident	nuclear accident	accident	0.80
Corrects	island accident	accident	mile island accident	accident	0.80
Corrects	natural human - induce event	event	human event	event	0.57
Corrects	Chernobyl accident	accident	Chernobyl accident	accident	1.00
Corrects	nuclear power plant accident	accident	nuclear power accident	accident	0.86
Corrects	surround area	area	surround area	area	1.00
Corrects	human health effect	effect	health effect	effect	0.80
Corrects	high risk	risk	high risk	risk	1.00
Corrects	nuclear accident	accident	nuclear accident	accident	1.00
Corrects	reactor accident	accident	reactor accident	accident	1.00
Corrects	Japanese people	people	Japanese people	people	1.00
Corrects	Chernobyl	accident	Chernobyl accident	accident	0.67
Corrects	radiation risk	risk	radiation risk	risk	1.00
Errors	nuclear accident material impact factor	factor	nuclear accident	accident	0.57
Errors	nuclear accident risk	risk	nuclear accident	accident	0.80
Errors	the Fukushima Daiichi nuclear disaster	measure	the Fukushima Daiichi nuclear power station	area	0.73
Errors	earthquake	event	earthquake risk	risk	0.67

First of all, many of the paired couples of the correct set are paired because the instance is basically expressed the exact same way. This is the case for the couple *health risk-risk* for example for which the ROUGE Score is equal to 1. But other instances are paired despite their expressions differ a little bit. For example, *nuclear power plant accident-accident* couple is considered as correct because it has been paired with the reference couple *nuclear power accident-accident* whose instance is not expressed with the same words but remains very close to the extracted one. This small difference is expressed by the ROUGE Score which is slightly smaller than 1 (0.85). Thus, despite this couple will not have the same weight as couples that perfectly match, it will still be considered as a correct couple.

Some extracted instances may be paired with a reference instance having a concept different from the one that is present in the extracted couple. These couples are classified as errors. For example, the instance *the Fukushima Daiichi nuclear disaster* is extracted as a *measure*. However, the reference dataset classes the paired instance *the Fukushima Daiichi nuclear power station* as an *area*. As instances present ROUGE Score above the threshold and their concepts differ, the relation detected between *the Fukushima Daiichi nuclear disaster* and the concept *measure* is considered as an error. Nevertheless, the E set underlines some limits of pairing instances from their expressions in different contexts. It happens for example that couples are assigned as errors even if the relation between the extracted instance and the concept remains relevant. For example the couple *earthquake-event* seems relevant. However, as the reference dataset contains the *earthquake risk-risk* couple which is paired with the *earthquake-event* couple, the former is wrongly considered as an error, which is mitigated by the associated ROUGE Score value (0.66). This limit is due to the fact that a single word, especially with the used extraction techniques can easily change the meaning of an instance, and especially the concept it should be linked to.

As explained before, being affected to set D does not mean for a couple that it represents an extraction that shouldn't happen as data sources are different. Two examples can be used to illustrate this. The couple *a comprehensive summary-organisation* is classified as a discovered relation as it appears in the extracted dataset but not in the reference dataset. As the relation does not seem to be relevant, it is probably a mistake and should contribute to false positives. The couple *flooding-event* is also classified as a discovered relation for the same reason. However, this relation is relevant, and the fact that it does not appear in reference data, does not mean that detecting it in extracted data is a mistake. Hence, it should contribute to true positives. The application on crisis management is different from the previous application in the sense that the size of the reference dataset is inferior to the size of extracted data (59 reference couples against 341 extracted couples). This choice has been made in order to show that the automated evaluation process proposed must be considered as a support for evaluation when only a few reference data are available. In order to make a decision concerning other couples – categorized as discovered – one should either (1) enlarge the reference dataset from other sources to cover more extracted couples or (2) use manual validation from a domain expert. The second solution has been applied in a previous paper [5] on several crisis management extracted datasets.

Similarly, being affected to set M does not mean for a couple that it is really a miss from the system. For example, the couple *hurricane-event* appears in the missed set as it is in the reference dataset but not in the extracted dataset. However, as the information about whether or not the instance *hurricane* does appear in the data is not available, it is hard to conclude if the couple should contribute to false negatives or not.

5.7. Further analysis on extraction results

Errors identified by the system have been explained and justified for each case. However, some errors still lie in the discovered dataset. Even if the goal of this paper is not to manually validate those errors, some investigation can be made in order to classify the different kinds of error that are made by the extraction system. These errors have been identified on both biochemistry and crisis management datasets, showing that they are not specific to a single domain.

As stated, the automated evaluation of extracted couples, besides its imperfection in the identification of errors can not classify all the extracted instances as correct or error if they do not appear (in a close enough expression) in each dataset, leading to missed and discovered datasets. Whereas most of the missed couples can be explained by the restrictive character of the designed rules or the simple absence of certain couples in the analyzed data, the discovered set should retain our attention. Looking closer into this set of couples, some type of errors have also been spotted leading to classification into three kinds of mistakes that the system makes:

- Incomplete instances: Some extraction often leads to incomplete instances. This kind of errors has been identified in both applications. They are principally due to the original generated parsing tree that does not connect all the terms characterizing an instance. As the evaluation system is flexible, incomplete instances can still be identified as correct but with a weaker score. Still, as the population framework allows the extraction of the raw sentence with the instance, it remains possible to correct manually an incomplete instance by referring to the extracted sentence.
- Semantically poor couples: A non-negligible part of the extracted couples is semantically correct, but brings little or no additional knowledge as they are extracted. Mostly, these couples are extracted through the $I = modifier + C$ pattern in which the modifier is either an adjective or a past participle. Once again, occurrences of these errors can be found in both applications. For instance, the terms *major protein* are extracted as an instance of the concept *protein*. Similarly, the terms *former population* might be recognized as an instance of the concept *population* even if the carried knowledge is not necessarily relevant.
- Inclusive concepts and polysemous concepts: As the issue concerned by concepts that include other concept has been evoked in the previous section, some concepts also lead to errors because of their polysemous character. This case can be encountered when common vocabulary is used to define a concept whose definition is really reserved to the domain of application. An assumption was made stating that a specific term is always used in the data with respect to the signification it has in the domain of interest. However, some exceptions exist. For example, the concept *Measure* refers to indicators in the crisis management domain but a part of instances extracted by the system refer to measures *taken* by institutions (*confinement measures*, *health measures*, *security measures*). The concerned couples end up as discovered not because they are semantically incorrect but because they are irrelevant regarding the definition of the concept.

6. Discussion

The results obtained opened new perspectives that can be discussed in order to propose further improvement of the evaluation system (section 6.1) and to underline how such a evaluation system is interesting for the addition of subjectivity in evaluation processes (section 6.2).

6.1. Towards a better sets content estimation

It is important to emphasize that the given definition of true positives, false positives, true negatives and false negatives adopts a pessimistic appreciation of extraction for the reasons discussed at the beginning of section 5.3. The optimistic counterpart would be to consider that:

- Couples contained in M can not be assimilated to false negatives as it could be assumed that these do not appear in the studied data.
- Couples contained in D are no longer assimilated to false positives but to true positives.

However, this view leads to a falsely maximum recall (equal to 1), and to a spiked precision, which will only very rarely be representative of the actual performance of the extraction. A better estimation of the precision can however be envisaged by not treating the case of the couples of the set D and by limiting the evaluation to the sets C and E to define true positives and false positives.

Moreover, the definition of M omits the couples that are not detected but whose instances have been detected in another couple, which makes the recall measure more distant from reality. Another definition of M, including these couples, would be to consider in some cases twice the same result of the system, first as an error from the point of view of the extracted data, then as a missing couple from the point of view of the reference data.

Finally, an automatic check of the presence of the instances of the couples of M in the text on which the extraction was carried out would be a solution which allows to better estimate the share of real false negatives in the set M. Thus, only the couples whose instances are present in the treated text can be considered in the definition of the false negatives. Nevertheless, process an automated check of the presence of a previously known entity in some data is a task in itself. From the difficulties encountered to identify reference instances within extracted datasets, it can be assumed that detecting the same instances in raw data will require some dedicated methodology as well.

6.2. Tackle expert subjectivity

An issue concerning data validation that has been introduced in section 5 is the subjectivity of the expert which has his own representation of the knowledge of a domain. A common way to attenuate this subjectivity, which exists in many validation processes, is to engage several experts in the process in order to build a consensus. In real-life situations, having several experts available to proceed to validation with a satisfying level of confidence consumes time and resources. Already existing annotated data, or knowledge bases are generally built under a consensus. Because of this consensus, they provide better objectiveness than a single expert validation. Thus, using this data as a reference – which is what the technique proposed in this article suggests – should also be in favor of a diminution of single expert bias during manual validation.

6.3. Making the automated evaluation reproducible

In the two given applications, the two respective reference datasets have the same final structure. However, they are extracted from different sources. In the crisis management application, the dataset is directly extracted from a previous extraction result stored in a graph database. From the database, the reference dataset can be built through a single request gathering all the validated instances of each concept registered in the database. In the biochemistry dataset, couples are extracted from annotated data, available as text files. Once some simple and purely technical work for the formatting into a list of couples has been done, these data can directly be used as a reference dataset. In the same spirit, the evaluation method is not attached to a specific relation extraction system. As its applicability has been shown with the relation extraction system presented in this paper it can be reused on any systems that extracts couples of hyponyms whether it is supervised or unsupervised, rule-based or supported by machine learning techniques. It also constitutes a solid base for the extension to the validation of other types of relations.

6.4. Improving the bootstrapping method

The bootstrapping method performs relatively poorly regarding the amount of knowledge deduced from generated rules. This explains partly the large amount of missed couples in the case of biological data. The issues encountered by the bootstrapping method are diverse. A big part of the deduced rules is a specification of larger rules. The algorithm has difficulties to get away from these original rules and explore to find new expressions of a relation. This observation is explainable by different aspects:

- The search of new patterns is restricted by the instances firstly identified with the unsupervised extraction system. The limited range covered by these relations consequently limits the group of identified relations that may re-appear somewhere else in the data. If this aspect may be tackled when harvesting larger sources of data, the size of the used data does not allow us to draw conclusions.
- The method as it is implemented, identifies a pattern only when the two terms (or groups of terms) are identified in the data exactly as they appear in the couple. Even if the lemmatized versions of the terms are used for identification, this does not take into consideration other expressions such as synonyms or co-references limiting the number of occurrence of each couple that should lead to a new pattern.
- The search of a pattern is limited to a sentence, removing the possibility to identify an instance in a given sentence that would be linked to a concept expressed in another sentence.

Those questions have not been investigated yet, but some insights can still be given. In some cases, the lack of initial identified relations could be corrected using external sources of data. However, this strategy is in conflict with the idea of an unsupervised system, that should itself bring the reference relation from an initial extraction. The issue linked to the expression of the terms could however be tackled using synonyms of the ontology concepts (and instances) or algorithms bringing co-reference analysis in addition to dependency parsing.

Some additional experiments have also been made relaxing the length constraint of the accepted patterns, but showed issues in term of computing time and relevance of the extracted patterns. When allowing bigger pattern to be selected, the relevant patterns are flooded by other non relevant patterns which affects the precision of the detection and the extraction itself.

7. Conclusion

In this paper, we presented (1) a rule-based approach for unsupervised relation extraction and (2) ontology population and a method for performance evaluation in the specific case of hyponymy relations extraction within unannotated textual data. The proposed method allows to estimate the performance of such an extraction model from reference data that are not necessarily linked to the data source from which the relations to be evaluated are extracted.

This definition of new performance evaluation methods is part of a more global approach of unsupervised ontology population whose general strategy has been presented. Especially, one of the methods allowing to carry out knowledge extraction from textual data has been detailled. Within the detailled method the emphasis is set on the first contribution of the paper which is the rule-based extraction system. This system proposes a redefinition of Hearst patterns in order to specify the extraction of relation depending on the domain of the targeted ontology.

The second contribution of the paper which is the proposed evaluation method, has been applied to two different sets of extracted data from distinct domains in order to show its genericity. Some errors and possible improvement of the method have been underlined through the analysis of the classification made in each application. Among them, it is relevant to focus on the limits of the system when dealing with uncommon reference data, in which the frontier between concepts is not clearly set. This observation opens new perspectives for the improvement of the evaluation system, especially concerning its ability to match couples of reference and extracted datasets. The presented equality measure showed good performances to match couples of extracted and reference data. However, the defined evaluation method is largely based on this equality measure which also has its own limitations. For example, the ROUGE Score does not take into account the semantic dimension between two instances and is then not sensible enough to word substitution that changes the meaning of an expression. In some cases, it can also lead to unwanted error detection. This performance measurement method must therefore be tested further, on bigger data sets, in order to prove its relevance. Nevertheless, other measures can be used or coupled with the ROUGE Score with the same performance evaluation method without affecting the global strategy.

Exploring the results of the evaluation system also conducted to the analysis of the extracted couples themselves. Some characteristics in the definition of the rules, in terms of domain coverage and precision have been highlighted. Notably, three classes of errors have been defined from the extraction results and detailed in the article.

Funding

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: Conceptualization, Data curation, Investigation, Methodology, Writing – original draft. **Anne-Marie Barthe-Delanoë:** Investigation, Supervision, Writing – review & editing. **Stéphane Négny:** Supervision, Writing – review & editing. **Jean-Marc Le Lann:** Supervision, Writing – review & editing.

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

Data will be made available on request.

Appendix A. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.ins.2023.01.150.

References

[1] J. Björne, T. Salakoski, Biomedical event extraction using convolutional neural networks and dependency parsing, in: Proceedings of the BioNLP 2018 Workshop, 2018, pp. 98–108.

[2] A. Boufrida, Z. Boufaida, Rule extraction from scientific texts: evaluation in the specialty of gynecology, J. King Saud Univ, Comput. Inf. Sci. (2020).

[3] D. Braun, A. Faber, A. Hernandez-Mendez, F. Matthes, Automatic relation extraction for building smart city ecosystems using dependency parsing, in: Proceedings of the 2nd Workshop on Natural Language for Artificial Intelligence, 2018.

[4] D. Cer, Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R.S. John, N. Constant, M. Guajardo-Céspedes, S. Yuan, C. Tar, et al., Universal sentence encoder, arXiv preprint, arXiv:1803.11175, 2018.

[5] Y. Chasseray, A.-M. Barthe-Delanoë, S. Négny, J.-M. Le Lann, Automated unsupervised ontology population system applied to crisis management domain, in: ISCRAM 2021-18th International Conference on Information Systems for Crisis Response and Management, 2021, pp. 968–981.

[6] Y. Chasseray, A.-M. Barthe-Delanoë, S. Négny, J.-M. Le Lann, A generic metamodel for data extraction and generic ontology population, J. Inf. Sci. (2021), https://doi.org/10.1177/0165551521989641.

[7] N. Chatterjee, N. Kaushik, RENT: regular expression and NLP-based term extraction scheme for agricultural domain, in: Proceedings of the International Conference on Data Engineering and Communication Technology, Springer, 2017, pp. 511–522.

[8] M. Chen, Y. Tian, X. Chen, Z. Xue, C. Zaniolo, On2Vec: embedding-based relation prediction for ontology population, in: Proceedings of the 2018 SIAM International Conference on Data Mining, 2018, pp. 315–323.

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

[10] L.N. De Silva, L. Jayaratne, WikiOnto: a system for semi-automatic extraction and modeling of ontologies using Wikipedia XML corpus, in: 2009 IEEE International Conference on Semantic Computing, IEEE, 2009, pp. 571–576.

[11] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, arXiv preprint, arXiv:1810.04805, 2018.

[12] N. Ding, G. Xu, Y. Chen, X. Wang, X. Han, P. Xie, H.-T. Zheng, Z. Liu, Few-NERD: a few-shot named entity recognition dataset, arXiv preprint, arXiv:2105.07464, 2021.

[13] T. Fawcett, An introduction to ROC analysis, Pattern Recognit. Lett. (2006) 27.

[14] F.N. Fote, A. Roukh, S. Mahmoudi, S.A. Mahmoudi, O. Debauche, Toward a big data knowledge-base management system for precision livestock farming, Proc. Comput. Sci. 177 (2020).
[15] A.L. Fraga, M. Vegetti, Semi-automated ontology generation process from industrial product data standards, in: III Simposio Argentino de Ontologías y sus Aplicaciones (SAOA)-JAIIO 46, Córdoba, 2017, 2017.
[16] Z. Geng, G. Chen, Y. Han, G. Lu, F. Li, Semantic relation extraction using sequential and tree-structured LSTM with attention, Inf. Sci. 509 (2020).
[17] T.R. Gruber, A translation approach to portable ontology specifications, Knowl. Acquis. 5 (1993).
[18] R.W. Hamming, Error detecting and error correcting codes, Bell Syst. Tech. J. (1950) 29.
[19] A.Z. Hassan, M.S. Vallabhajosyula, T. Pedersen, UMDuluth-CS8761 at SemEval-2018 task 9: hypernym discovery using Hearst patterns, co-occurrence frequencies and word embeddings, arXiv preprint, arXiv:1805.10271, 2018.
[20] M.A. Hearst, Automatic acquisition of hyponyms from large text corpora, in: Coling 1992 volume 2: The 15th International Conference on Computational Linguistics, 1992.
[21] A. Huet, R. Pinquié, P. Véron, A. Mallet, F. Segonds, CACDA: a knowledge graph for a context-aware cognitive design assistant, Comput. Ind. 125 (2021).
[22] L. Jin, L. Song, Y. Zhang, K. Xu, W.-y. Ma, D. Yu, Relation extraction exploiting full dependency forests, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, 2020, pp. 8034–8041.
[23] K.S. Jones, A statistical interpretation of term specificity and its application in retrieval, J. Doc. (1972).
[24] A.C. Khadir, H. Aliane, A. Guessoum, Ontology learning: grand tour and challenges, Comput. Sci. Rev. 39 (2021) 100339.
[25] S. Kübler, R. McDonald, J. Nivre, Dependency parsing, Synth. Lect. Hum. Lang. Technol. 1 (2009) 1–127.
[26] T.K. Landauer, P.W. Foltz, D. Laham, An introduction to latent semantic analysis, Discourse Process. (1998) 25.
[27] V.I. Levenshtein, Binary codes capable of correcting deletions, insertions, and reversals, Sov. Phys. Dokl. 10 (1966) 707–710, Soviet Union.
[28] C.-Y. Lin, ROUGE: a package for automatic evaluation of summaries, in: Text Summarization Branches Out, 2004, pp. 74–81.
[29] K. Liu, W.R. Hogan, R.S. Crowley, Natural language processing methods and systems for biomedical ontology learning, J. Biomed. Inform. 44 (2011).
[30] P. Lomov, M. Malozemova, M. Shishaev, Training and application of neural-network language model for ontology population, in: Proceedings of the Computational Methods in Systems and Software, Springer, 2020, pp. 919–926.
[31] L. Luo, Z. Yang, M. Cao, L. Wang, Y. Zhang, H. Lin, A neural network-based joint learning approach for biomedical entity and relation extraction from biomedical literature, J. Biomed. Inform. 103 (2020).
[32] K.A. Nguyen, M. Köper, S.S.i. Walde, N.T. Vu, Hierarchical embeddings for hypernymy detection and directionality, arXiv preprint, arXiv:1707.07273, 2017.
[33] M.-S. Paukkeri, A.P. García-Plaza, V. Fresno, R.M. Unanue, T. Honkela, Learning a taxonomy from a set of text documents, Appl. Soft Comput. 12 (2012).
[34] M. Pennacchiotti, P. Pantel, A bootstrapping algorithm for automatically harvesting semantic relations, in: Proceedings of the Fifth International Workshop on Inference in Computational Semantics, ICoS-5, 2006.
[35] S. Roller, D. Kiela, M. Nickel, Hearst patterns revisited: automatic hypernym detection from large text corpora, arXiv preprint, arXiv:1806.03191, 2018.
[36] L.M. Sanagavarapu, V. Iyer, Y.R. Reddy, OntoEnricher: a deep learning approach for ontology enrichment from unstructured text, arXiv preprint, arXiv:2102.04081, 2021.
[37] D. Sanchez, A. Moreno, Creating ontologies from web documents, in: Recent Advances in Artificial Intelligence Research and Development, 2004, p. 113.
[38] M. Shardlow, N. Nguyen, G. Owen, C. O'Donovan, A. Leach, J. McNaught, S. Turner, S. Ananiadou, 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, 2018, pp. 280–285.
[39] T. Thongkrau, P. Lalitrojwong, OntoPOP: an ontology population system for the semantic web, IEICE Trans. Inf. Syst. 95 (2012).
[40] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, arXiv preprint, arXiv:1706.03762, 2017.
[41] W.E. Winkler, Overview of record linkage and current research directions, in: Bureau of the Census, Citeseer, 2006.
[42] Y. Wu, S. Zhao, W. Li, Phrase2Vec: phrase embedding based on parsing, Inf. Sci. 517 (2020) 100–127.
[43] Y. Zhang, H. Lin, Z. Yang, J. Wang, S. Zhang, Y. Sun, L. Yang, A hybrid model based on neural networks for biomedical relation extraction, J. Biomed. Inform. 81 (2018).