Non-ontological sources transformations for Ontology Engineering: knowledge acquisition using redundancy

Fabien Amarger[1,2], Jean-Pierre Chanet[2], Ollivier Haemmerlé[1], Nathalie Hernandez[1], and Catherine Roussey[2]

[1] IRIT, UMR 5505, UT2J, Dlépartement de Mathlématiques-Informatique, 5 alllées Antonio Machado, F-31058 Toulouse Cedex, France - firstname.lastname@univ-tlse2.fr
[2] UR TSCF, Irstea, 9 av. Blaise Pascal CS 20085, 63172 Aubiere, France - firstname.lastname@irstea.fr

Abstract. Non-ontological sources like thesauri or taxonomies are already used as input in ontology development process. Some of them are also published on the LOD. Reusing this type of sources to build a Knowledge Base (KB) is not an easy task. The ontology developer has to face different syntax and different modelling goals. We propose in this paper a methodology to transform several non-ontological sources into a single KB. We claim that our methodology improves the quality of the final KB because we take in account: (1) the quality of the sources, (2) the redundancy of the knowledge extracted from sources in order to discover the consensual knowledge and (3) Ontology Design Patterns (ODP) in order to guide the transformation process.
We have evaluated our methodology on the agriculture domain by creating a knowledge base on cereals taxonomy from three non-ontological sources.

Keywords: Knowledge acquisition, Non-Ontological sources, trust, Ontology Design Pattern, consensus, quality, agriculture

1 Introduction

In many fields, domain specific information is distributed on the Web as structured data (such as databases or thesauri) gathered for a specific usage. End-users are often lost when facing this amount of data as they have to seek for available sources, analyse their quality, retrieve specific information from each of them and compare them. Alongside, the Linked Open Data (LOD) initiative aims at linking and facilitating querying on available data. Approaches such as [17, 20] have been proposed to formalise existing sources, define vocabularies to describe them and publish them on the LOD. However, approaches are still needed in order to help end-users to collect and access knowledge to achieve a specific task in specialised domains.

This is for example the case in the Agriculture domain which is out of a phase. Over recent decades, agricultural practices have evolved considerably due to various constraints: societal and environmental issues, regulatory framework, and climate changes. Meanwhile, the role of data in agriculture has also changed significantly: first used for traceability and food safety data, now they contribute directly to change agricultural practices. Agriculture practices should evolve to drastically reduce pesticide usage. Farmers and agronomists must rethink agricultural practices according to experiments. Before testing new practices, it is necessary to accumulate knowledge about plant development (or any topic related to agricultural practices like chemistry product or meteorology) and provide as knowledge as possible. In agriculture, the LOD is relatively undeveloped and exploitable data are available as structured data.

In this paper, we propose a method for building knowledge bases addressing a specific issue covering end-users' need from non-ontologicial sources such as thesauri or classifications. The main idea is to design an ontology module representing the knowledge needed by end-users and to enrich it automatically with data extracted from existing sources. The originality of our proposition is to exploit the consensus found in existing sources in order to increase the trust of the elements added to the knowledge base. The paper is organised as follows. A state of the art is presented in section 2. Section 3 focuses on the main ideas of our proposition. Section 4 details our methodology and section 5 presents experiments that have been carried out on a real life case in the agricultural domain.

2 State of the Art

2.1 Reusing non-ontological sources

NOS (Non-Ontological Sources) are already used to build domain ontologies. They can be used in different processes of ontology engineering methods.

1. One of the first processes is knowledge extraction or knowledge discovery. This process delimits the scope of the domain ontology and discovers key concepts. It uses NOS to extract a glossary of terms. We can cite the UPON methodology [1] or SMOL[8].
2. Part of NOS can be reengineered in order to produce ontology modules like in the method [20] of the NEON methodology [18]. These methods combine knowledge discovery and knowledge organisation processes, i.e. not only terms are extracted from NOS but also conceptual structure or facts (like concepts hierarchy or data property associated with concepts or instances).
3. NOS can be used to document ontologies and add new labels. For example in the ontology engineering method proposed in [13], thesaurus is used as a pivot language to connect set of ontologies and associated updated labels with ontology concepts.

The approach we propose is a NOS reengineered method. Thus, in the scope of this paper we will focus on NOS representing conceptual structures such

as thesauri. We claim that the representation of the conceptual structure is context dependant. It depends on the final usage of the application for which the conceptual structure is built. The translation of the representation from a NOS to an ontology should be adapted.

Because of the lack of formalisation of NOS, the transformation from this kind of sources into a knowledge base is particularly complex. The extraction of the ontological objects (potential elements for the knowledge base source [8]) is quite similar in all methodologies, but the analysis of the elements nature (is it an instance, a class, a relation? and which relation is it exactly?) is different. This is the disambiguation. All methodologies agree on that the disambiguation depends on the domain and on the exploited source. [20] uses an external resource to disambiguate the extraction. This requires being sure that the resource can be trusted on the specific domain.

Then all methodologies to transform NOS into knowledge bases could imply some errors on the disambiguation process so the result is not necessarily of good quality. The final knowledge base quality is directly dependant of the input source quality. If the source quality is good, then the disambiguation process is easier. So we have to define the source quality to estimate the result quality.

2.2 Definition of source quality

In order to determine how much of a source is reusable, we have to specify some quality criteria to consider. Depending on the conceptual representation type of the source, the work to determine criteria are different. For example, [19] suggest some metrics to evaluate thesaurus, while[14] present a method to evaluate the knowledge base. But the most exhaustive works are about databases, [11] sums up criteria from all available work on this domain. Some works tried to generalize criteria for all implementation sources type [9]. All studies reach an agreement on the complexity of being absolutely exhaustive. [3] proposes a system, which uses the source quality definition. This system considers a sub-part of available criteria (such as reputation, citation count, publication date, experimental design, ...) because of the difficulty to be exhaustive.

2.3 Ontological object trust

Extracting ontological objects from various NOS with different qualities requires a consideration of trust on these elements. Several trust definitions in computer science and semantic web are presented in [2]. The one which corresponds the most to our purpose is:

"Trust of a party A to a party B for a service X is the measurable belief of A in that B behaves dependable for a specified period within a specified context (in relation to service X)."

Consider A as the user who wants to create a knowledge base, B as a source and X the extraction process. In this definition, trust is about a source B, with the extraction process X, which generates ontological objects candidates with a score associated. This definition is very suitable to our purpose because

they consider that a trust is specific for a period, a context and a service. This corresponds to the fact that the trust on a source is variable depending on the objective of the project, the time and the source itself.

Using multiple sources to extract ontological objects leads to a more precise consideration of trust than using only one source. Finding the same ontological object from several sources increases the trust score of this object using redundancy. As shown on [4] this redundancy based computation of the trust score is more effective than classic approaches. But these works do not consider the quality of the sources, all sources bring the same amount of trust. We claim that a source brings different amount of trust depending on its quality.

3 Main ideas - Module and consensual trust for knowledge extraction

3.1 Methodology

Due to its genericity we choose to work on the Neon methodology. As far as we know, Neon is the only methodology that helps building modular ontologies [15] in order to improve their re-usability and understanding. Neon proposes a set of nine methods that involve different processes for collaboratively building modular ontologies. Each method is dedicated to a specific situation, for example to reuse Ontology Design Patterns (ODP).

Our proposition consists in linking two ontology engineering methods (called scenario 7 and scenario 2) of the Neon methodology. As seen in figure 1 we start with the method called scenario 7. This method reuses some design patterns matching our requirements and ends-up with a module which is the result of ODP fusion and merging (see sub-section 3.2). We modify this scenario by inserting the scnerario 2 at the 6th activity. This scenario enrich the module with some ontological objects extracted from non ontological sources. Our goal is to

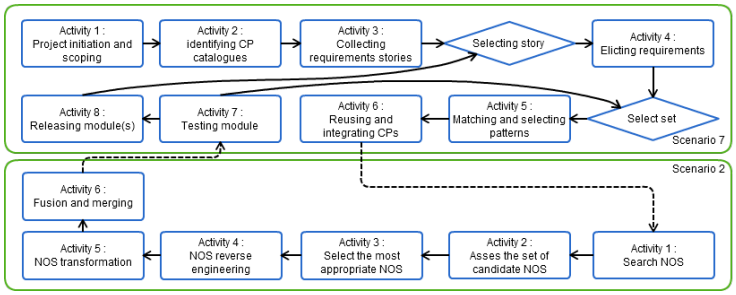

Fig. 1. NeOn scenarii fusion

build a knowledge base with less human intervention and with a good modelisation because of the use of design patterns. To perform the second part of this method (enrich the module with NOS), a transformation pattern for each source is proposed (see sub-section 4.2).

3.2 Module

The current best practice to create an ontology is to reuse ODP. We follow the method [7] in order to generate modules. An example of one of our generated modules is AgronomicTaxon [16] illustrated in the figure 2 which has been manually built for a specific task (representing the taxonomic classification). The module is composed of *owl:classes* and defines the set of object properties that may exist between them. The aim of our work is to enrich this module with ontological objects extracted from the different sources, i.e. we want to add new instances and new relationships between instances and all the labels available for each of them. We can also enrich the module with new classes. These new ontological objects extracted from sources are specialisations or instantiations of the objects composing the module.

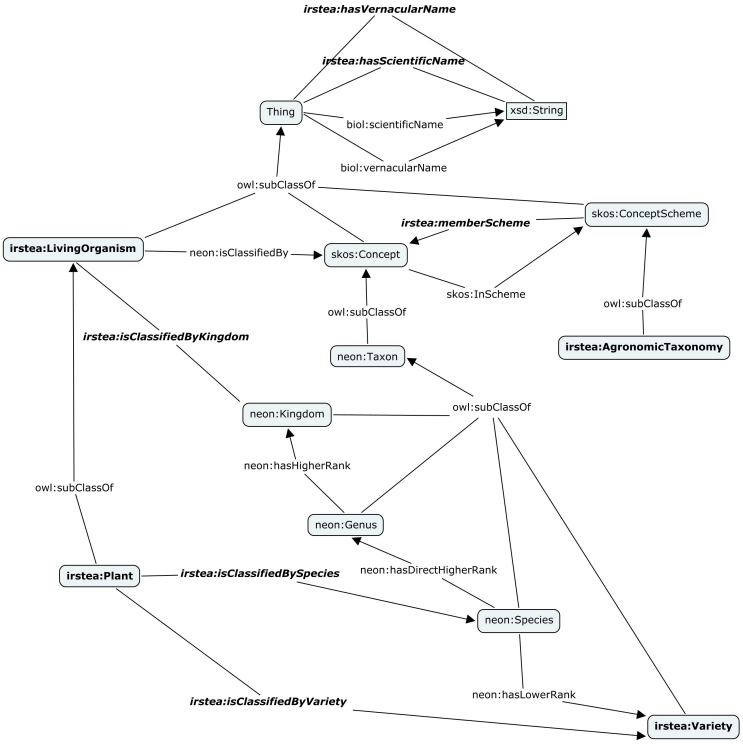

Fig. 2. AgronomicTaxon

The AgronomicTaxon module models living organism taxonomy. All the taxon types wanted in our knowledge base are defined in the module as *owl:class*, child of the *neon:taxon* class. For example we only focus on the seven most known taxon types: kingdom, genus, family, order, class, phylum and species. We need to define all the taxa, the instances of the class taxon. Using a module when building an ontology from a NOS is useful for two main reasons: (1) it defines the domain limits that the ontology should cover and help extracting the relevant

pieces of knowledge from sources, (2) it guides the design. As each source can have its own conceptualization, with some specificities, specialising the module helps combining the knowledge in a uniform representation defined for the targeted functionality. Enriching a module with knowledge extracted from several sources, especially from large sized sources, can lead to errors as we present in the state of the art section. To determine whether an element is wrong or not, we must analyse the extracted knowledge from all sources and identify what is consensual.

3.3 Consensual trust for knowledge extraction

We multiply the sources of knowledge to find out where they agree to increase the confidence of an ontological object. So if an ontological object is present in several sources, we associate a better trust score with it. Then, we can trust this ontological object more than if it was present only once. This idea of trust is directly associated with the source quality idea as we presented in the state of the art. Finally, each ontological object extracted has a score which represents the value of the confidence on it. This value is the aggregation from the sources quality and the number of sources which contain the ontological object.

4 Knowledge extraction process

4.1 Process overview

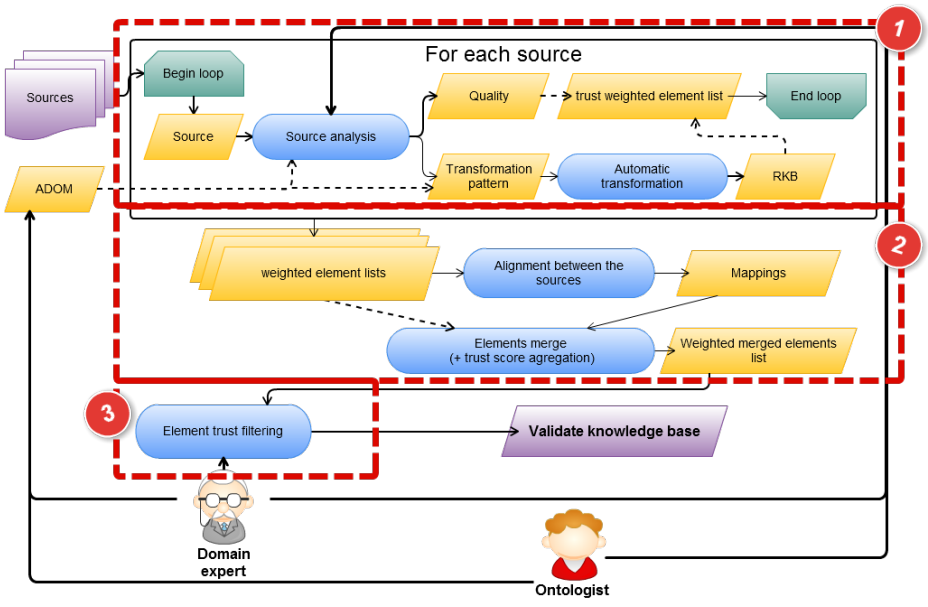

Fig. 3. Process overview

We present here an overview of the process to transform several sources into a knowledge base according to a module using consensual truth. This process is divided in three different parts:

1 - Source analyzing: During this step, the domain expert and the ontologist work together to manually define for each source two outputs: (1) the source quality and (2) how to transform this source in order to obtain an automatically generated knowledge base (cf section 4.2). The source quality is used to determine how much we can trust an ontological object extracted automatically from this source;

2 - Align and merge knowledge bases: This step aligns and merges automatically the knowledge bases resulting from the first step. The idea here is to obtain links between the similar ontological objects from the different sources and merge them into a single one. The merge part also aggregates the trust score of each similar ontological object to determine the trust score of the final object;

3 - Filtering the result: The aggregated trust score is used to reduce the number of elements the domain expert will have to validate. The system should be able to validate automatically some ontological objects that will be considered as trustable.

4.2 Source analyzing

The first step of the process consists in analyzing each source in order to determine if this is a reliable source and how the system can transform it into a generic format. This step is currently done manually with a domain expert as it aims a representing his knowledge need and requires his expertise both on the considered source and the task to achieve.

Transformation pattern

As generally each NOS has its specific format and correspond to specific modelization choices, it becomes difficult to determine a generic way to transform a source. The state of the art showed that the most convenient way to determine how to transform them is to define a pattern for each special feature (modelization or implementation). We use the same idea as in the Villazon-Terrazas' work [20], a pattern based transformation. Where Villazon-Terrazas's work determines a pattern for a specific modelization, we want to determine a pattern for a specific source and for a specific knowledge extraction need. That requires more work, because the created patterns are not really reusable but this is more effective because of the specific features of each source and each task a knowledge base is built for. But it is still possible to use a Villazon-Terrazas' transformation pattern in our methodology. What we call a transformation pattern is an algorithm which will be used by the system to automatically generate a knowledge base. As we saw in section 2, results of the automatic transformation methods are, in most cases, partly wrong. The goal here is not to obtain a perfect result

at this step of the process but to have a first step done to work on the same kind of implementation and modelization format. We use here an OWL[3] format to output the result. Simple algorithms are preferred because all the complex treatment will be done on a next step using the strength of the large number of source to validate or not ontological objects. The first step to create a transformation pattern is to define manually an alignment between relevant elements from the module considered and elements from the sources. These mappings will be used by the algorithm. For example, if we consider AGROVOC[4] as a source to enrich the module AgronomicTaxon we can define the algorithm 1. The mappings are used during the *rdf:type* assignment for each instance. In this algorithm, the element **Plants** is one of the results of the mappings to focus the extraction on a specific part of the source. We simplify the problem here by extracting only

Algorithm 1 Transformation Pattern : AGROVOC for AgronomicTaxon

```
for all termGroup UNDER Plants do
  instTaxon ← newInstance(termGroup);
  instTaxon.setScientificName(termGroup.getPreferedLabel());
  instTaxon.setVernacularName(termGroup.getAlternativeLabel());
  if termGroup.existsRelation("hasTaxonomicLevel") then
    instTaxon.setType(termGroup.getRelation("hasTaxonomicLevel");
  else
    instTaxon.setType("Taxon");
  end if
  if termGroup.existsRelations("broader") then
    instTaxon.setRelation("hasHigherRank", termGroup.getRelation("broader"));
  end if
end for
```

instances, labels and the "hasHigherRank" relation but we can easily extend the process with the extraction of other kind of ontological objects.

Source quality definition

As discussed in the state of the art, the number of criteria allowing the definition of the source quality is too large to use all of them. Defining exhaustively the source quality is still an open problem. For now we decide to use three criteria:

The source reputation represents the usage of the source by many people and the reference usage by the experts;

The source freshness represents the last time the source has been updated and if it is updated often;

The source adequacy with the targeted task (module similarity) represents the similarity between the source and the desired knowledge base represented by the considered module.

[3] http://www.w3.org/TR/2012/REC-owl2-overview-20121211/

[4] A multilingual thesaurus about agriculture built by the FAO with more than 40'000 terms - http://aims.fao.org/standards/agrovoc

These criteria are the most suitable for the project because they describe the main characteristics needed when we want to use a source in this kind of process.

The source quality is then defined using these criteria evaluation. But each criterion has not the same importance for a specific project. For some project the freshness is much more important than the reputation or the module similarity. That is why we use weights for each criterion. At the beginning of the project, the domain expert has to define which criterion is more important and which is not by specifying the integer weight for each criterion. At the end we can compute the source quality value by a weighted sum with the formula 1.

$$SourceQual(S) = \frac{\sum_{i=0}^{nbCriteria} weigth(Crit_i) * value(S, Crit_i)}{\sum_{i=0}^{nbCriteria} weigth(Crit_i)} \quad (1)$$

At this step we have each source transformed into a knowledge base and a quality value associated. The next step is to find out the knowledge that are similar in several sources.

4.3 Align and merge Knowledge bases

Aligning system

Aligning two knowledge bases consists in finding which ontological objects are equivalent in two knowledge bases. This is a large research area [5] and a lot of methods have been proposed. In order to determine how we can align the knowledge bases, we have looked at the last OAEI challenge[5] and especially at the instance matching results[6] [10], because we want to map all kind of ontological objects. We chose to use LogMap [12] because of their good results and because the source code is available online[7].This system is really easy to use, especially thanks to the SEALS API[8] which all the OAEI participants have to use. That means if, next year, a system is better than this one, it will be easy to change the alignment system. For each couple of knowledge bases generated automatically, we use LogMap to get the mappings between them. The LogMap system returns all the mappings available and the mapping trust score associated (between 0 and 1). At the end of this step we have a weighted mapping list for each couple of knowledge bases generated automatically. We use these mappings to merge the sources.

Merge policy

Let us define a mapping m as a triplet $< e_i, e_j, s_{ij} >$ such as:

$e_i \in KB_i$**:** is an ontological object belonging to KB_i ,
$e_j \in KB_j$**:** is another ontological object belonging to KB_j ($KB_j \neq KB_i$),

[5] Ontology Alignment Evaluation Initiative - http://oaei.ontologymatching.org/2013/
[6] http://www.instancematching.org/oaei/imei2013/results.html
[7] https://code.google.com/p/logmap-matcher/
[8] http://www.seals-project.eu/

s_{ij}**:** is the similarity degree between e_i and e_j.

We define a function called $degree(e_i, e_j)$ from $KB_i \times KB_j$ to $[0, 1]$. Where $s_{ij} = degree(e_i, e_j)$ is the similarity trust between e_i and e_j given by the alignment process (LogMap) and 0 if there is no alignment. We define a candidate c as a set of mappings that share common ontological objects. Each ontological object, belonging to a candidate c, should belong to distinct knowledge bases. Let consider the example in Figure 4 with three knowledge bases, each one containing two elements. The *cross* elements of each are aligned so we can defind the candidate c as :

$$c = [< e_1, e_2, s_{12} >, < e_2, e_3, s_{23} >, < e_1, e_3, s_{13} >]$$
$$e_1 \in KB_1, e_2 \in KB_2, e_3 \in KB_3$$

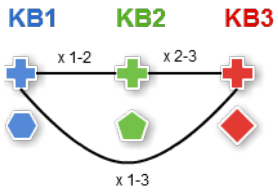

Fig. 4. Aligning example

We define $dim(c)$ as the number of KB involved in c. (three in the example figure 4). For each candidate c we compute a trust score aggregating the quality of the different sources involved in c. We define $trust(c)$ from the candidate c to $[0, 1]$ as follows:

$$trust(c) = \sigma\left(\left(\sum_{i=1}^{dim(c)} SourceQual(KB_i)\right) * \left(\sum_{i=1}^{dim(c)} \sum_{j=i+1}^{dim(c)} degree(e_i, e_j)\right)\right)$$

This equation first sums the source quality of each KB involved in the candidate, and then sums all the trust scores of the alignments involved. This formula is surrounded by the sigmoid (σ) function, which is specified in the formula:

$$\sigma(x) = \frac{1}{1 + \exp^{5-(3x/2)}}$$

This sigmoid function is used to normalize the result and to represent our intuition of confidence. This distribution is convenient to represent the confidence because of the smoothness in the extreme values. A small increase of the confidence is not sufficient to trust a candidate but after a certain limit we can trust easily. This sigmoid function represents exactly that sensation. If we apply this equation on the example shown in (figure 4), considering ic_1 as the instance candidate from the alignment for each *cross* on all the knowledge bases, we get the following result:

$$trust(ic_1) = \sigma((SourceQual(KB_1) + SourceQual(KB_2) + SourceQual(KB_3))$$
$$* (degree(e1, e2) + degree(e2, e3) + degree(e1, e3)))$$

For each candidate generated we search if there is a relation "hasHigherRank", extracted on the transformation pattern, between the candidate and another one. If there is one, we generate a candidate of the ontological object with the same idea as for the instance candidate previously defined. The formula is the same as $trust(c)$ but the alignment trust sum is replaced by the $trust(c)$ of the object instance candidates involved on the relation. For example, if there is a relation candidate rc between the instance candidate ic_1 as subject and instance candidate ic_2 as object, and this relation has been found in two knowledge bases (KB_1 and KB_2) then there is:

$$trust(rc) = \sigma((SourceQual(KB1) + SourceQual(KB2)) * trust(ic2))$$

Then, the trust score of ic_1 is changed by:

$$trust(ic_1) = \sigma\left(\left(\sum_{i=1}^{dim(ic_1)} SourceQual(KB_i)\right) * \left(\sum_{i=1}^{dim(ic_1)} \sum_{j=i+1}^{dim(ic_1)} degree(e_i, e_j)\right) + trust(rc)\right)$$

This is because if an instance candidate is involved in a relation candidate, then we have a trust value increased on this instance candidate depending on the relation trust score. This is the core part of our project since, during this step, we agregate all the ontological objects extracted from the sources and especially we agregate the trust score. This score represents the evaluation of the trust on the candidate of the final ontological object. Then we can decide if we will keep it or reject it because of its trust score. This function $trust(c)$ is really important for the result of our project.

4.4 Filtering the result

The filtering step is facilitated by the trust score associated on each candidate of ontological object extracted on the previous step. The trust score represents the consensus value between all the knowledge bases considered. In that case we can define a minimum and a maximum thresholds, defined between 0 and 1. The minimum threshold is used as a limit below which the candidate is automatically rejected without any validation, wheras the maximum threshold is used as a limit over which the candidate is automatically accepted. Between these two limits, the domain expert has to validate manually each candidate. This filtering step reduces the number of candidates that the domain expert has to validate manually because only candidates between the two thresholds need to be checked.

5 Experiments

As discussed in the introduction, there is a lack of KB on the agricultural domain. We want to fill this gap by creating a KB for this domain and especially for the observation of bio-aggressors attacks on crops and techniques to fight against. This KB will be used to annotate the french corpus *Les Bulletins de*

Santé du Végétal [9] which lists bio-aggressors attacks in France. We start with a module about plant classification (*AgronomicTaxon*) and with the following three sources:

Agrovoc: [10]Multi-lingual thesaurus with more than 40,000 terms,
TaxRef: [11]Taxonomic referential with 80,000 taxa created by the french "Muséeum national d'histoire naturelle",
NCBI Taxonomy: [12] Taxonomy created by the National Center for Biotechnology Information (NCBI) of the United States with 1,000,000 taxa.

With this module we want to extract plants taxonomic classification. After this step we will work on other module to extract attacks observations and defence techniques. We will link them to obtain a KB with enough knowledge to annotate the corpus cited before. In order to validate the result manually with cereals experts, we extract subparts of the cereals taxonomic classification. We focus the extraction on the Triticum (wheat) and the Aegilops (wild wheat) taxa. We define transformation patterns to extract instances of *Taxon* from the three different sources. For each instance we consider the relation *hasHigherRank* which is the hierarchical relation used to describe the taxonomy hierarchy. This relation is defined in AgronomicTaxon. We consider also the *Type* relation to define at which level the taxon is (Specy, Genus, Family, ...) from the AgronomicTaxon module.

We chose these three sources because of their complementarity. First NCBI is the source with the most taxa. It is considered by experts to be the most up-to-date source but this include potential errors and there are only few labels. Alongside, Agrovoc contains labels in several languages with distinctions between scientific labels and vernacular ones but less taxa than NCBI and with a quality often criticized[17]. TaxRef overcomes this drawback and is considered as a national reference in agronomic classification. It is developed by the National Museum of Natural History in France, but the number of taxa is limited by the data verification process. Combining these three sources is very suitable because we combine the taxa quantity (NCBI), with labels quantity (Agrovoc) and the assurance of quality (TaxRef).

To validate our approach, we asked to three domain experts to analyze the three knowledge bases extracted automatically from the sources. The experts had to determine which ontological object were well represented and in the scope of the knowledge base needed. Our purpose is to validate the intuition that the knowledge kept from each source are the common knowledge between each of these sources. To do so, first we analyzed if the experts had a consensual point of view on the knowledge which should be kept from each source. Then we evaluated the quality of the ontological objects extracted by our approach by comparing them to those validated by the three experts.

[9] http://www.mp.chambagri.fr/-Bulletin-Sante-du-vegetal-.html
[10] http://aims.fao.org/standards/agrovoc/about
[11] http://inpn.mnhn.fr/programme/referentiel-taxonomique-taxref
[12] http://www.ncbi.nlm.nih.gov/

5.1 Evaluation of the consensus intuition

To validate our candidates with the experts, we first have to know if there was a consensual knowledge on this domain. To do so we computed a ratio between the number of experts agree and the number of validations. We consider that they agree when at least two experts validate the ontological object and the third one voted on the *Don't know* option. We get a consensual ratio of **0.82**. We also computed the Fleiss Kappa[6] score on the results. Then we get the Fleiss Kappa of **0.69**. These two scores show that the experts agree, most of the time, on the result of the ontological objects classification. So we can use this consensual aspect of the domain and use the experts validation as a gold standard to validate candidates.

5.2 Evaluation of the quality

Then, we computed the precision, recall and f-measure for the Triticum and the Aegilops from the instance candidates and the relation candidates. We extracted only candidates with a trust score greater than 0.9 to consider them as keepable candidate. For example, this threshold rejects candidates found only in NCBI and Agrovoc. In this extraction we computed the precision using the ratio between the expert validated candidates (when the experts agree, as described in the previous paragraph) and the number of candidates kept. The recall is computed by the ratio between the experts validated ontological objects in the candidates from the prototype and the number of experts validated ontological objects. The f-measure is the well-known aggregation between these two scores. The results

	Triticum	Aegilops
Precision Instance	0.95	1
Recall Instance	0.62	0.36
F-Measure Instance	0.75	0.53
Precision Relation	0.94	1
Recall Relation	0.31	0.31
F-Measure Relation	0.47	0.47

Table 1. F-Measure results

shown on the table 1, show us that the recall is often quite low but the precision is quite high. That means we generate only few relevant candidates but most of them are validated by the experts. This is what we expected when setting a threshold at 0.9. The low value of the recall can be explained by the high value of the maximum thresholds (0.9) and the number of validated elements on NCBI. NCBI has eight times more taxa than the two others sources, so all taxa can't be aligned with other sources.

6 Conclusion and future works

In this article, we proposed two major ideas aiming at helping the knowledge extraction from several sources. The first one consists in using a module which

allows to focus the domain of the project, in order to extract only the interesting part from the source. This module also helps to normalize modeling and improve the consistency between the elements extracted. The second idea is to use source quality and consensus in order to compute a trust score for each ontological object extracted. You can trust a source more than another one because of their quality values and an extracted ontological object is probably more trustable if you can find it in several sources. Based on these two ideas, our system produces a set of candidates weighted with a trust value. This helps the validation at the end of the process because some candidates could be validated (or rejected) automatically, by using different filtering thresholds. The consensus helps the system to determine the trust score onto which we can filter the candidates.

In this paper we developed details about the two first parts of the process: the source analyzing and the alignment and merging of KB. We will focus our next works on the filtering of the results, in order to answer to several problems we observed. The first problem is the treatment of the contradictions. Currently all the candidates are considered and can be accepted, even if there is a conflict. To solve such conflicts, it could be possible to use the argumentation theory associated with the trust score to manage the candidate selection. One of the best practices is to keep the provenance of data to track where they come from. This could be integrated to our prototype to keep the candidate provenance and allows to add another source afterward. Thanks that only the modifications given by the new source will be computed and not all the process. The W3C offers a provenance ontology[13] and a perspective to our work is to extend this ontology to add our specific information. We want also to work on another sub-domain than the plants taxonomy classification. We planned to work on the attacks from bio-aggressors using the module CultivatedPlant[14] with a database from the Arvalis[15] and others sources which still have to be defined.

References

1. Roberto Navigli Antonio De Nicola, Michele Missikoff. A proposal for a unified process for ontology building: Upon. In *Database and Expert Systems Applications*, pages 655–664, 2005.
2. Donovan Artz and Yolanda Gil. A survey of trust in computer science and the semantic web. *Web Semantics: Science, Services and Agents on the World Wide Web*, pages 58–71, 2007.
3. Patrice Buche, Catherine Dervin, Ollivier Haemmerlé, and Rallou Thomopoulos. Fuzzy querying of incomplete, imprecise, and heterogeneously structured data in the relational model using ontologies and rules. *Fuzzy Systems*, pages 373–383, 2005.
4. Doug Downey, Oren Etzioni, and Stephen Soderland. A probabilistic model of redundancy in information extraction. Technical report, DTIC Document, 2006.

[13] http://www.w3.org/TR/prov-o/
[14] https://sites.google.com/site/agriontology/home/irstea/cultivatedplant
[15] http://www.arvalis-infos.fr

5. Jérôme Euzenat, Pavel Shvaiko, et al. *Ontology matching*, volume 18. Springer, 2007.
6. Joseph L Fleiss, Bruce Levin, and Myunghee Cho Paik. *Statistical methods for rates and proportions*. John Wiley & Sons, 1984.
7. Aldo Gangemi and Valentina Presutti. Ontology design patterns. *Handbook on Ontologies*, pages 221–243, 2009.
8. Richard Gil and Maria J. Martín-Bautista. Smol: a systemic methodology for ontology learning from heterogeneous sources. *Journal of Intelligent Information Systems*, pages 415–455, 2014.
9. Yolanda Gil and Donovan Artz. Towards content trust of web resources. *Web Semantics: Science, Services and Agents on the World Wide Web*, pages 227–239, 2007.
10. Bernardo Cuenca Grau, Zlatan Dragisic, Kai Eckert, Jérôme Euzenat, Alfio Ferrara, Roger Granada, Valentina Ivanova, Ernesto Jiménez-Ruiz, Andreas Oskar Kempf, Patrick Lambrix, et al. Results of the ontology alignment evaluation initiative 2013. In *ISWC workshop on ontology matching (OM)*, pages 61–100, 2013.
11. Vimuthki Jayawardene, Shazia Sadiq, and Marta Indulska. An analysis of data quality dimensions. Technical report, 2013.
12. Ernesto Jiménez-Ruiz, Bernardo Cuenca Grau, Yujiao Zhou, and Ian Horrocks. Large-scale interactive ontology matching: Algorithms and implementation. In *European Conference on Artificial Intelligence*, pages 444–449, 2012.
13. Antonio Jimeno-Yepes, Ernesto Jiménez-Ruiz, Rafael Berlanga Llavori, and Dietrich Rebholz-Schuhmann. Reuse of terminological resources for efficient ontological engineering in life sciences. *BMC Bioinformatics*, page 4, 2009.
14. María Poveda-Villalón, Mari Carmen Suárez-Figueroa, and Asunción Gómez-Pérez. Validating ontologies with oops! In *Knowledge Engineering and Knowledge Management*, pages 267–281. Springer, 2012.
15. Alan L Rector. Modularisation of domain ontologies implemented in description logics and related formalisms including owl. In *Knowledge capture*, pages 121–128. ACM, 2003.
16. Catherine Roussey, Jean-Pierre Chanet, Vincent Cellier, and Fabien Amarger. Agronomic taxon. In *Workshop on Open Data*, page 5. ACM, 2013.
17. D. Soergel, B. Lauser, A. Liang, F. Fisseha, J. Keizer, and S. Katz. Reengineering thesauri for new applications: The AGROVOC example. *Journal of Digital Information*, 4, 2004.
18. Mari Carmen Suárez-Figueroa, Asunción Gómez-Pérez, Enrico Motta, and Aldo Gangemi. *Ontology engineering in a networked world*. Springer, 2012.
19. Osma Suominen and Christian Mader. Assessing and improving the quality of skos vocabularies. *Journal on Data Semantics*, pages 47–73, 2013.
20. Boris Villazón-Terrazas, Mari Carmen Suárez-Figueroa, and Asunción Gómez-Pérez. A pattern-based method for re-engineering non-ontological resources into ontologies. pages 27–63, 2010.