ISSN 2395-8618

Identification of axiomatic relations from unstructured texts using named entity recognition

Ana B. Rios-Alvarado, Ivan Lopez-Arevalo, and Edgar Tello-Leal

***Abstract*—Domain ontologies facilitate the organization, sharing and reuse of domain knowledge. The construction of ontologies from text deals with the extraction of concepts and relations from a text collection. A huge challenge is the learning of more expressive ontologies which includes relations such as disjointness or equivalence between classes. In this paper, we propose a method for recognition of named entities, which operates on the levels of instance and class. Firstly, at the instance level, using a named entity recognition tool named entities from unstructured texts are extracted. In addition, the type and subtype of the extracted named entity are identified. Secondly, at the class level, for each class a set of instances that allow characterizing the class is associated. Then, using the type and the set of instances of each class, the proposed method can identify the axiomatic relation. The different axiomatic relations that approach identifies can be *subClassOf*, *disjointWith*, and *equivalentClass*. The evaluation of the method for named entity recognition proposed was performed using a data set of 3542 English text documents.**

***Index Terms*—Knowledge acquisition, ontologies, text processing**

I. Introduction

The use of information and communication technologies have motivated an exponential growth in the available information. This growth is not only present on web resources, but it also can be seen in organizations. For example, in an organization, documents represent a significant source of collective expertise (*know-how*) and the most of the data are in unstructured text format. For instance, the number of business emails sent and received per user per day totals 122 emails per day[1]. In order to store, retrieve, or infer knowledge from this information, it is necessary to represent it using a conceptual schema. This can be achieved by means ontologies. Ontologies are formal vocabularies of terms, often shared by a community of users [1]. Ontologies facilitate the organization, sharing, and reuse of domain knowledge, they also are one of the key technologies for the Semantic Web and its current success.

Ontology learning from text consists in deriving high-level concepts and relations on the basis of the words appearing in the text [2]. To carry out this process, textual documents are an important source of knowledge. Moreover, in the recent years, the availability of unstructured textual information has increased, which can serve to extract useful knowledge. In many areas, such as medicine, bioinformatics, and finance, the main benefits of using ontologies for knowledge modeling is the ability to infer new knowledge that allows the development of more realistic applications, which requires the inclusion of more expressive elements, such as disjointness or equivalence relations. Axioms involving semantic features that can provide expressivity to ontologies [3]. Consequently, the addition of such relations allows the implementation of applications based on reasoning tasks, such as ontology classification and query answering.

In the context of languages for Semantic Web (for example OWL-DL), an axiom is an assertion in a logical form. All axioms together comprise the overall theory that the ontology describes in its domain of application. Taking into account the elements of the ontology, there are three types of axioms: 1) *class expression axioms*, which refer to general restrictions between classes, for example, the *subClassOf* relation between the *SoccerClub* and *SportTeam* classes, or *disjointWith* relation between the *City* and *SoccerClub* classes; 2) *properties* allow to define the attributes or facts associated with the members of classes or specific instances, for example, the relation *birthPlace* between *Place* and *Person* classes or the relation *birthYear* between *Person* class and `xsd:integer`; and 3) *assertions* on individuals commonly called *facts*, for example, the relation between individuals with the same characteristics establishes a particular property between them, such as *Ronaldo* `owl:sameAs` *Ronaldo Luís Nazário de Lima*. In particular, OWL-DL gives the formal syntax to represent the axioms above described in the ontology. The disjointness of classes can be expressed using the `owl:disjointWith` constructor. This relation guarantees that an individual, as member of one class, cannot be simultaneously an instance of a specified other class. Similarly, the constructor `owl:equivalentClass` is used to indicate that two classes have precisely the same instances. The obtaining of instances for each class is a key step in the identification of subsumption, disjointness or equivalence relations.

This paper presents a method based on named entity recognition from unstructured text to identify class expression axioms. A named entity is an information unit such as the name of a person, an organization, a location, a brand, a product, or a numeric expression (time, date, money, and percent) as can be found in text. The presented approach starts with the detection of named entities. Subsequently, at the class level, for each class a set of instances that allow characterizing the class are identified and associated. In a complementary way, the sentences where the instances and their corresponding type of class appear are analyzed. Consequently, the context relation and the *instanceOf* relations based on entity extraction task, determines one of the following relations between classes: *subClassOf*, *disjointWith*, or *equivalentClass*. This is

[1]The Radicati Group, Inc, Email Statistics Report, 2015-2019 www.radicati.com

ISSN 2395-8618

possible due to the use of schema types from AlchemyAPI or OpenCalais have also been collected in an ontology called NERD (Named Entity Recognition Disambiguation)[2]. Finally, the evaluation of the method was performed using a data set of 3542 English documents in the Football domain, allowing evaluate the identification of the *instanceOf* relation, and evaluate the learning axioms. In [4] has been reported the results for a set of documents in Tourist domain.

The rest of the paper is structured as follows. In Section 2, a brief description of the work related to generation of axioms is presented. Next, in Section 3 the method to identify class expression axioms is described. In Section 4, the experiments carried out are presented and discussed. Finally, in Section 5, we provide some conclusions.

II. Related work

In order to provide a higher level of expressiveness to learned ontologies, several approaches have been proposed for extending logical properties of the modeled knowledge in an unsupervised or automatic way. According to the the type of axioms, works such as [5], [6], and [7] are focused on class expression axioms. The tool named LEDA [5] permits the automated generation of disjointness axioms based on machine learning classification. The classifier, which determines disjointness for any given pair of classes, is trained based on a gold standard baseline of disjoint axioms manually created. Zhang *et al.* [6] proposed an unsupervised method for minning equivalent relations from Linked Data. It consists of two components: 1) a measure of equivalency between pairs of relations of a concept and 2) a clustering process to group equivalent relations. Ma *et al.* [7] introduced an approach to discover disjointness between two concepts. In this work, the task of association rule minning is to generate patterns like the form $A \rightarrow \neg B$, and then transform them to disjointness axiom "A `owl:disjointWith` B". On the other hand, Sánchez *et al.* [8] presented an approach for discovering object properties. Their method is based on natural language processing techniques, linguistic patterns and statistical analyses performed at a Web-scale to extract and evaluate semantic evidences from textual resources. In [9] and [10] the approaches are related work to assertions or inference rules acquisition. Völker *et al.* [9] presented the methodology named LExO. The first step of the methodology is analyzing the syntactic structure of an input sentence. The resulting dependency tree is transformed into a set of OWL axioms (concept inclusion, transitivity, role inclusion, role assertions, concept assertions, and individual equalities) by means of manually engineered transformation rules. Li and Sima [10] proposed an ontology mining approach, where the ontology axioms are obtained through statistical measures by running SPARQL queries on Linked Data.

The above approaches do not examine how to determine what classes are relevant in an automatic way for getting axioms neither do they consider the individuals as part of the extensional definition of a class. In order to get axioms, by taking into account the evidence of named entities in domain-specific text, we propose to resolve the following question: Does the *instanceOf(named entity, class)* relation provide evidence for an axiomatic relation? To address this question, the named entities have been identified by a Named Entity Recognition (NER) tool and subsequently, *subClassOf*, *disjointWith*, and *equivalentClass* relations are established. The NER aims to identify meaningful segments in input text and categorize them into pre-defined semantic classes such as the names of people, locations and organizations.

We assumed that a taxonomy structure exists and it represents the domain of the texts. Following a method from specific to general, the approach involves identifying individuals, which are instances of some class. Such classes belong to a taxonomic structure, which is at the core of the ontology. Figure 1 shows that the instance level corresponds to the leaves in a taxonomic tree structure and the class level to the branches. The difference between one class and another is that its set of leaves is different and therefore it can be characterized as a separate (disjoint) class, otherwise if the set of leaves is very similar, then it can be characterized as an equivalent class. For example, in the instance level, the set of leaves for *Country* class includes *Brazil*, *Germany*, and *Denmark* as members, but the set of leaves for *SoccerFederation* class contains *FIFA*, *CONMEBOL*, and *UEFA* members. Then, *Country* class and *SoccerFederation* class are disjoint. Thus, the collection of named entities provides the members for a specific class, and defines a class in an extensional manner.

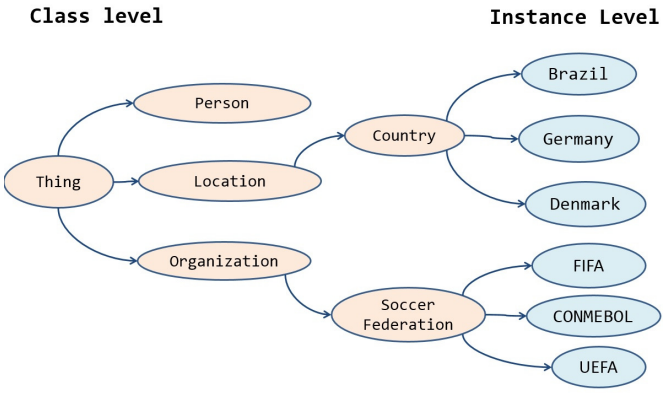

Fig. 1. Example of ontology for Sports domain

III. A method for acquisition of axioms

The proposed method starts at the instance level, where an NER tool extracts the named entities from input text. Later, at the class level, each class has a set of instances associated with it that characterize it. The NER tool provides a set of types (type/subtype) associated to each named entity. Using the type and the linguistic context of each class, an axiomatic relation is identified. Figure 2 shows the general overview of the proposed steps to extract axioms. This method consists of a bottom-up approach and it follows the next steps:

1) Identification of instances: An NER tool obtains the named entities from text. The named entities can correspond to one of the following types (defined by the tool):

[2]Available: http://nerd.eurecom.fr/ontology

ISSN 2395-8618

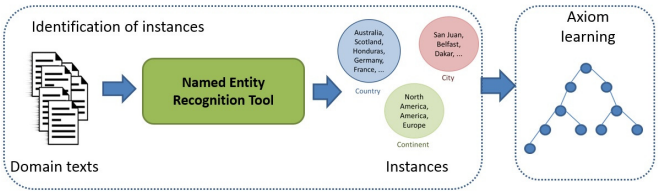

Fig. 2. The proposed method for identification of axioms

Person, Organization, Location, Country, or Quantity among others. The NER tools exploit the Linked Data[3] principles, which consists of a unique global identifier defines an entity. Such referenced identifier provides useful information about the corresponding resources and links to other relevant identifiers. Later, the relations of type *instanceOf(named entity, class)* between a named entity and a class are obtained by two methods: 1) the given type from the NER tool and 2) the context where the named entity and its class co-occurs.

2) Axiom learning. The sentences where a set of instances and its corresponding class occur are grouped to determine if there exists a relation between the contexts of two classes. A part-of-speech (POS) tagger and a syntactic parser are used to get the linguistic context (i.e., representative elements such as nouns, verbs, or adjectives and their grammatical relations). The linguistic context supports the identification of relations based on entities used to derive one of the following axioms: *disjointWith* or *equivalentClass*.
At the class level, the *subClassOf* relation represents one of the main axioms, which structures the set of classes into a taxonomy where a higher class is more general than a lower class. We propose the use of NER and linguistic context as an additional approach for identifying *subClassOf* relations in text.

IV. Experiments and Results

For our experiments, we used the Smart Web Football dataset used by Jiang and Tan [11], which consists of 3,542 English documents. It covers a list of 2295 classes, 1459 individuals, and 633 taxonomy relations. The measures used for the evaluation are precision, recall, and F-measure.

A. Identification of instances

In this stage, the objective was to evaluate the identification of the *instanceOf* relation using AlchemyAPI and OpenCalais tools. These tools execute the named entity recognition task and define a taxonomy of types. The comparison was made on 185 *instanceOf* relations that were manually annotated. According to the evaluation, AlchemyAPI had better precision than OpenCalais in this task. More in detail, Table I presents the performance of AlchemyAPI and OpenCalais for the identification of instances belonging to these classes: *Country*, *Person*, *City*, and *Company*. The obtained results were compared manually with 70 *instanceOf* relations from the test dataset manually annotated. In most cases, AlchemyAPI showed the best precision.

TABLE I
Performance NER tools - identified instances by class

Class	Tool	Precision	Recall	F Measure
Country	AlchemyAPI	0.4529	0.4900	0.4707
	OpenCalais	0.4000	0.5000	0.4444
Person	AlchemyAPI	0.7331	0.8582	0.7907
	OpenCalais	0.6500	0.7000	0.6740
City	AlchemyAPI	0.5678	0.4000	0.4693
	OpenCalais	0.5234	0.3550	0.4230
Company	AlchemyAPI	0.3333	0.2667	0.2963
	OpenCalais	0.2480	0.3000	0.2715

TABLE II
Examples of sentences where *instanceOf* relation ocurrs

Sentence	Lexical Pattern
Messi is an Argentine professional *footballer* who plays as a forward for Spanish club Barcelona and the Argentina national team.	<NE> is a <NP>
I have actually wanted to be a proffessional *goalkeeper, like Iker Casillas* from Spain.	<NP> like <NE>
Eight *players including Brian McBride*, *Claudio Rayner*, and *Brad Friedel*	<NP> including <NE> {, <NE>, ... and <NE>}

In addition, using the context, we can see that instances of different classes appear in the same sentence, i.e. they co-occur. For extracting relations, the linguistic context for each of the extracted named entity was analyzed. The Table II shows examples of sentences with patterns that identify the *instanceOf* relation, where <NE> is a named entity and <NP> is a noun phrase. In the first example, *Messi* is an instance of the *footballer* class and the pattern associated is <NE> is a <NP>. In the second example, *Iker Casillas* is an instance of the *goalkepper* class. In this case, the pattern associated is <NP> like <NE>. For the third example, the instances are *Brian McBride*, *Claudio Rayner*, and *Brad Friedel* for the class called *player*.

B. Axiom learning

In this section, we present a description on the experiments to identify *subClassOf*, *disjointWith*, and *equivalentClass* relation.

The NER tool used for this was AlchemyAPI because it shows the best precision in obtaining instances. AlchemyAPI obtains 16 types of classes and 62 subtypes on a sample corpus with 541 files from the Smart Web Football corpus. A human team was asked to evaluate all extracted subtype relation, which gave a precision of 73.58% for the extracted relations based on AlchemyAPI identified subtypes-types representing the football domain. The Table III shows some examples of relations correctly identified.

A *disjointWith* relation states that one class has not an instance member in common with another class. For learning the disjoint relationship between two classes, we consider named

[3]http://www.w3.org/DesignIssues/LinkedData.html

ISSN 2395-8618

TABLE III
EXAMPLES OF LEARNED *types-subtypes* RELATIONS

Type	**Subtype**
Organization	*SoccerClub*, *FootballTeam*, *FootballOrganization*
Company	*FootballAssociation*, *SportsAssociation*
Person	*FootballPlayer*, *FootballManager*
Sport	*AwardDiscipline*
Region	*Location, Country*

TABLE IV
EXAMPLES OF LEARNED *disjointClass* RELATIONS AND ITS NAMED ENTITIES

$class1$/$class2$	$class1$**'s NE**	$class2$**'s NE**
SportingEvent/ Organization	*World Cup*, *Nations Cup*, *Olympics*	*Arsenal*, *FIFA*, *Champions League*, *Glasgow Rangers*, *East Asian Football Federation*
Country/Organization	*Italy*, *Japan*, *Iraq*, *Germany*, *United States*, *France*, ...	*Arsenal*, *FIFA*, *Champions League*, *Glasgow Rangers*, *East Asian Football Federation*
City/SportingEvent	*Kuwaits*, *Cologne*, *Aruba*, *Liverpool*, *Caracas*, *Miami*, *Madrid*, ...	*World Cup*, *Nations Cup*, *Olympics*
City/Person	*Kuwaits*, *Cologne*, *Aruba*, *Liverpool*, *Caracas*, *Miami*, *Madrid*, ...	*Jacques Santini*, *Patrick Mboma*, *Edwin van der Sar*, *Hidetoshi Nakata*, ...

entities that co-occur in the same context. For each NER ($class_1$, $class_2$) duple, the list of instances was compared. If there is not a common named entity between the two classes then the *disjointWith(*$class_1$, $class_2$*)* relation is established. To illustrate the evaluation of *disjointWith* relation extraction, it was used a sample corpus with 541 files. A number of 120 duple ($class_1$, $class_2$) were obtained. According to the evaluation of the human team, where it was evaluated if obtained duple has disjoint relation between $class_1$ and $class_2$, 102 of the relationships correspond correctly to *disjointWith (*$class_1$, $class_2$*)* and the rest of them (18) have some other relation. As a result, the precision was 85.00% for the learned disjoint relations. Some examples of learned disjoint relations between classes are the *SportingEvent* and *Organization* classes as well as the *Country* and *Organization* classes, *City* and *SportingEvent* classes, and the *City* and *Person* classes. However, the *Organization* and *Company* are not necessary disjoint classes. Even although according to NER tool results, the set of instances were very different between *Organization* and *Company*, according to human expert the classes meet in a *subClassOf* relation. The Table IV shows some disjoint relations learned and their corresponding named entities where it is clear that their set of named entities is disjoint.

In a particular case, the sets of named entities associated with *City* class and *SoccerClub* class are very similar, but these class are disjoint although they share elements.

The *equivalentClass* relation is established between two classes when the class descriptions include the same set of individuals. It is important to mention that class equality means that the classes have the same intensional meaning i.e. denote the same concept. For learning *equivalentClass* relation, two ontologies were considered and for each ontology class its set of instances obtained by two different NER tools were compared, if the set of instances between two different classes is highly similar then an *equivalentClass(*$class_1$, $class_2$*)* relation can be established. Highly similar means that almost the total of named entities detected by the NER tool is the same in both classes, that is because the identification of instances depends on the precision of the NER tool. In this case, using the same sample corpus with 541 files, the AlchemyAPI and OpenCalais tools identify 16 and 17 classes, respectively. However, only 32 duple (AlchemyAPI : $class_1$, OpenCalais : $class_2$) of the total (272) have overlap between their set of instances. For example, *AlchemyAPI : Organization* / *OpenCalais:Organization* and *AlchemyAPI : Country* / *OpenCalais : Country* can clearly be determined a equivalence relationship between them. In contrast, the classes *AlchemyAPI : Organization* / *OpenCalais : Company* or *AlchemyAPI : Person* / *OpenCalais : Holiday* which have similar individuals but they are not equivalent. According to the evaluation of the human team, 24 of the relationships correspond correctly to *equivalentClass(*$class_1$, $class_2$*)* and the rest of them have some other relation. As a result, the precision was 75.00% for the learned *equivalentClass* relations. The Table V shows some examples of learned duples, where *AAPI* and *OC* correspond to *AlchemyAPI* ontology and *OpenCalais* ontology, respectively.

TABLE V
EXAMPLES OF EQUIVALENT CLASSES

$class1$	$class2$	*equivalent class*	**other relation**
AAPI:Organization	*OC:Organization*	*	
AAPI:Country	*OC:Country*	*	
AAPI:Sports	*OC:SportsGame*	*	
AAPI:Health Condition	*OC:Medical Condition*	*	
AAPI:Organization	*OC:Company*		*

V. CONCLUSIONS

The approach described in this paper is based on identifying named entities as class' members and comparing their set of instances to establish axiomatic relations *subClassOf*, *disjointWith* and *equivalentClass*. Our approach is unsupervised and the identified relationships can enrich ontologies lack of expressiveness. New technologies in NER tools based on Linked Data can be useful in the process of extracting axioms.

According to the experiments, we observed that the identified instances that belong to a specific class could be considered as the extensional definition of this class and then it is described by the named entities associated to it. However, the method must take into account the fact that the incorrect identification of instances can derive erroneous axiomatic relations. For example, other relations such as *subClassOf* and *partOf* were learned instead as a *disjointWith* relation, or as *equivalentClass* relation. One of the main difficulties lies with

ISSN 2395-8618

resolving ambiguity in named entities. In such case, other tools could be exploited for named entity disambiguation task.

In the experiments, one of the main dificulties lies with ambiguity. Further work will be focus on more experiments for adding other resources and evaluating the similarity of classes. Also, new experiments will consider a comparison with other approaches.

REFERENCES

[1] I. Horrocks, "Tool support for ontology engineering," in *Foundations for the Web of Information and Services*, 2011, pp. 103–112.

[2] P. Cimiano, *Ontology Learning and Population from Text: Algorithms, Evaluation and Applications*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[3] J. Völker and S. Rudolph, "Lexico-logical acquisition of owl - dl axioms," in *Formal Concept Analysis*, ser. Lecture Notes in Computer Science, R. Medina and S. Obiedkov, Eds. Springer Berlin / Heidelberg, 2008, vol. 4933, pp. 62–77. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-78137-0_5

[4] A. Rios-Alvarado, I. Lopez-Arevalo, and E. Tello-Leal, "The acquisition of axioms for ontology learning using named entities," *IEEE Latin America Transactions*, vol. 14, no. 5, pp. 2498–2503, 2016.

[5] J. Völker, D. Vrandečić, Y. Sure, and A. Hotho, "Learning disjointness," in *The Semantic Web: Research and Applications*, ser. Lecture Notes in Computer Science, E. Franconi, M. Kifer, and W. May, Eds. Springer Berlin Heidelberg, 2007, vol. 4519, pp. 175–189. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-72667-8_14

[6] Z. Zhang, E. Blomqvist, I. Augenstein, F. Ciravegna, and A. L. Gentile, "Mining equivalent relations from linked data," *Proc. of the 51st Annual Meeting of the Association for Computational Linguistics*, vol. 2, pp. 289–293, 2013.

[7] Y. Ma, H. Gao, T. Wu, and G. Qi, "Learning disjointness axioms with association rule mining and its application to inconsistency detection of linked data," in *The Semantic Web and Web Science*. Springer, 2014, pp. 29–41.

[8] D. Sánchez, A. Moreno, and L. Del Vasto-Terrientes, "Learning relation axioms from text: An automatic Web-based approach," *Expert Systems with Applications*, vol. 39, no. 5, pp. 5792–5805, 2012.

[9] J. Völker, P. Hitzler, and P. Cimiano, "Acquisition of owl dl axioms from lexical resources," in *The Semantic Web: Research and Applications*, ser. Lecture Notes in Computer Science, E. Franconi, M. Kifer, and W. May, Eds. Springer Berlin / Heidelberg, 2007, vol. 4519, pp. 670–685. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-72667-8_47

[10] H. Li and Q. Sima, "Parallel mining of OWL 2 EL ontology from large linked datasets," *Knowledge-Based Systems*, vol. 84, pp. 10–17, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.knosys.2015.03.023

[11] X. Jiang and A.-H. Tan, "Crctol: A semantic-based domain ontology learning system," *J. Am. Soc. Inf. Sci. Technol.*, vol. 61, no. 1, pp. 150–168, Jan. 2010.