Expert Systems With Applications 113 (2018) 339–355

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

OpenIE-based approach for Knowledge Graph construction from text

Jose L. Martinez-Rodriguez [a,*], Ivan Lopez-Arevalo [a], Ana B. Rios-Alvarado [b]

[a] Cinvestav-Tamaulipas, Victoria, Mexico
[b] Facultad de Ingeniería y Ciencias, Universidad Autónoma de Tamaulipas, Victoria, Mexico

ARTICLE INFO

Article history:
Received 24 January 2018
Revised 18 June 2018
Accepted 6 July 2018
Available online 10 July 2018

Keywords:
Knowledge Graph
Semantic Web representation
Fact extraction
Relation Extraction
RDF events

ABSTRACT

Transforming unstructured text into a formal representation is an important goal of the Semantic Web in order to facilitate the integration and retrieval of information. The construction of Knowledge Graphs (KGs) pursues such an idea, where named entities (real world things) and their relations are extracted from text. In recent years, many approaches for the construction of KGs have been proposed by exploiting Discourse Analysis, Semantic Frames, or Machine Learning algorithms with existing Semantic Web data. Although such approaches are useful for processing taxonomies and connecting beliefs, they provide several linguistic descriptions, which lead to semantic data heterogeneity and thus, complicating data consumption. Moreover, Open Information Extraction (OpenIE) approaches have been slightly explored for the construction of KGs, which provide binary relations representing atomic units of information that could simplify the querying and representation of data. In this paper, we propose an approach to generate KGs using binary relations produced by an OpenIE approach. For such purpose, we present strategies for favoring the extraction and linking of named entities with KG individuals, and additionally, their association with grammatical units that lead to producing more coherent facts. We also provide decisions for selecting the extracted information elements for creating potentially useful RDF triples for the KG. Our results demonstrate that the integration of information extraction units with grammatical structures provides a better understanding of proposition-based representations provided by OpenIE for supporting the construction of KGs.

1. Introduction

Information consumed every day by people in a variety of services such as supermarkets, banks, libraries, and web search engines is internally stored in a structured fashion to be efficiently queried and transformed. However, nearly 95% of data is unstructured (Tanwar, Duggal, & Khatri, 2015), which means that valuable information has not been explored and that would be useful in applications such as user preferences (satisfiability), merchandising, or demographic movements, to mention a few. Hence, such data need to be transformed into a structured format in order to be handled and processed by applications and users. The Web has been a valuable data source for several tasks such as NLP, Information Extraction, Machine Learning, among others. However, much of such data is unstructured, in the form of text, which is unfeasible or very expensive to process due to its large scale and heterogeneity. In this sense, one of the aims of the Semantic Web (Daconta, Obrst, & Smith, 2003) is to extract and formally represent information by leveraging formal data interchange formats, standards, and technologies of the Web.

In recent years, many data publishers have relied on the benefits provided by the Semantic Web for quickly publishing, parsing and processing data by machines. This development has been partially supported by the Linked Open Data (LOD) initiative[1] with more than 80 billion published RDF triples.[2] Such data has been mainly extracted from (semi-) structured sources (e.g., relational databases, meta-data, Wikipedia infoboxes, HTML tables). Nevertheless, a huge amount of information from the Web is mainly stated as plain text (without any structure or description), and translating it into a structured format requires text manipulation tasks provided by areas such as Natural Language Processing (NLP), Information Extraction (IE), and Information Retrieval (IR). With the support of such areas, two main elements are typically ex-

* Corresponding author.
E-mail addresses: lmartinez@tamps.cinvestav.mx (J.L. Martinez-Rodriguez), ilopez@tamps.cinvestav.mx (I. Lopez-Arevalo), arios@uat.edu.mx (A.B. Rios-Alvarado).

https://doi.org/10.1016/j.eswa.2018.07.017
0957-4174/

[1] LODstats http://stats.lod2.eu. All URLs in this paper were last accessed on 2018/05/18.
[2] An RDF triple is a unit of information composed of three elements: Subject-Predicate-Object.

tracted and semantically annotated from text: named entities[3] and semantic relations between them.[4] The extraction of such elements and their representation on the Semantic Web are the main components of a task that we coin as *Relation Extraction and Linking* (REL). Broadly speaking, the output of the REL task is a (RDF) graph that, in the context of the Semantic Web, is known as a *Knowledge Graph (KG)* (Ehrlinger & Wöß, 2016), where nodes refer to named entities and edges to the semantic relation between them.[5] The difficulty of constructing Knowledge Graphs by REL-based approaches relies on the large scale and heterogeneity of the text (as seen in the Web) and linguistic problems such as detection of synonymy (e.g., words *cat* and *kitty*) and ambiguity (e.g., identify the word *orange* as fruit or as color), to mention a few. Moreover, there are no defined standards that indicate the specific information to be extracted and formally represented, which is a difficult decision even for a knowledge domain expert. However, we identified three kinds of REL approaches that deal (partially) with such issues: Discourse-based, Distant Supervised-based, and OpenIE-based.

First, *Discourse-based* approaches analyze the use of language in terms of written and spoken communication structures (Corcoglioniti, Rospocher, & Aprosio, 2016; Exner & Nugues, 2012), whose goal is the unsupervised understanding of text. Such approaches are useful for constructing taxonomies and building connections between propositions extracted from text. However, their use involves the representation of text with a combination of logical implications and linguistic structures that may complicate the data consumption (i.e., querying and parsing data).

Second, *Distant Supervised-based* approaches train a machine learning algorithm with information from a KG. Thus, similar facts to those represented in a KG are extracted from text. Although such approaches often provides high levels of precision, their use is limited to a closed world assumption (everything not in the training KG is false) and to the often complex adjustment of parameters used by such algorithms.

Finally, and the focus of this work, *OpenIE-based*[6] approaches (Dutta, Meilicke, & Stuckenschmidt, 2014; 2015) obtain propositions (usually binary relations) with no restriction of a domain and without requiring training data. Although OpenIE has received less attention (in the context of RDF triple extractions) than the above-mentioned approaches, it is a useful technique that allows a semantic representation through *predicative* statements (clauses where the verb is the core of the relation) that may be helpful for shallow questions (e.g., *who directed the Star Wars movie?*). However, REL approaches based on OpenIE present some issues regarding the recognition of entities, properties and the representation of binary relations. For example, given the following running example "The clinician –Dr. Gregory House– diagnosed a cancer patient in New York City", we can explain the following aspects:

- Entity Linking. Named entities extracted from text must be linked to resources of a KG.[7] The often reduced number of extractions returned by existing systems for such purpose (further explained in this paper as *Entity Extraction and Linking*) limit the number of relations that can be represented. For example, in the running example, we may only obtain and link to a KB (DBpedia) the entity clinician(dbr:clinician)[8] but additional entities (such as Cancer(dbr:Cancer), or New York(dbr:New_York_City)) are necessary to represent a complete RDF triple. In consequence, REL approaches (Gangemi, Presutti, Recupero, & Nuzzolese, 2017) are sometimes prone to automatically generate IRI identifiers for the named entities, resulting in differences of meaning and interpretation (data heterogeneity).
- Entity selection. Several entities might appear as the subject or object of the relation. However, there are no criteria for picking a single one. For example, given the binary relation (extracted from the running example) diagnosed(Dr. Gregory House, cancer patient in New York) and the entities cancer(dbr:Cancer) and patient(dbr:patient), the selection of one entity over the other to represent the object of such relation is not a straightforward task. Existing REL approaches (Dutta et al., 2014; Exner & Nugues, 2012) address this aspect by rules that restrict the selection of specific types of entities. However, such restrictions limit the result to a particular kind of relations to be extracted.
- Property linking. In the previous binary relation, the relation phrase "diagnosed" should also be identified with an IRI from a resource associated to an ontology[9] or a KG. Existing approaches (Dutta, Meilicke, & Stuckenschmidt, 2015; Exner & Nugues, 2012) map relation phrases to KG properties through generated rules and text similarity measures. However, such mappings may not always exist and thus, an alternative solution needs to be stated.
- Representation. OpenIE-based REL approaches typically represent a binary relation with only one triple. However, due to the number of involved elements, a single RDF triple is often insufficient for representing the information and thus, some elements cannot be represented. In the case of the extracted relation previously presented, the named entity New York City would not be represented within the triple if we select Gregory House as subject and cancer as object (because the same triple element cannot contain two resources).

According to the above mentioned issues, in this paper, we propose an approach based on OpenIE relations for the construction of KGs from plain text in English. Similar to existing systems, our approach consists of the integration of NLP/IE tools and Semantic Web technologies for processing and representing elements of text. Moreover, we propose some linguistic associations that allow for extracting and integrating semantic relation elements and named entities on RDF triples. We opted for OpenIE before other approaches because it provides binary relations that constitute atomic units of information used to convey facts (with no need of training and additional configurations), which can simplify the querying and presentation of data.

As a motivating example, given the running example sentence, our approach would be able to extract and represent binary relations like diagnosed(Dr. Gregory House, cancer patient) using an RDF graph model as presented in Fig. 1. We rely on an *n*-ary representation (several triples that convey ideas of the same statement), where the binary relation is broadly represented through the node (`ex:d1evt5`); the relation phrase is described by the node (`pmn:pb215-diagnose.01`), which denotes the

[3] Named entities refer to real world things and/or concepts denoted by a proper name, such as persons, organizations, places, etc.

[4] A semantic relation refers to the relationship between two or more named entities, where the most typical is the binary relation (two named entities connected through a relation phrase) with the form relationPhrase(Subject,Object).

[5] In the context of the Semantic Web, nodes and edges of a KG are known as resources and must be individually identified through Internationalized Resource Identifiers (IRI) and retrieved (dereferenced) through the HTTP protocol.

[6] Open Information Extraction (OpenIE).

[7] We represent a mention of a named entity together with its IRI identifier with the form mention(IRI).

[8] For space reasons, we use IRI prefixes (namespaces) in accordance with the service hosted at http://prefix.cc, where dbr:Cancer represent a contraction for the IRI http://dbpedia.org/resource/Cancer.

[9] An ontology refers to *terminological knowledge*. In the Semantic Web, ontologies define the properties and classes of a domain that can be used to construct RDF triples.

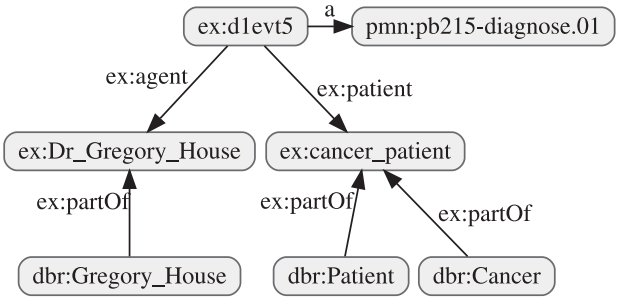

Fig. 1. Example of RDF graph representation.

main action or *event* of the semantic relation and which was disambiguated over resources of a lexical database to get the correct sense. With the correct sense of the relation, we are able to represent the *causer* of the action (denoted by the property `ex:agent`) and the *undergoer* of the action (denoted by the property `ex:patient`) through a semantic analysis. Finally, entities that belong to the same grammatical unit of information (particularly Nominal Phrases (NP))[10] are associated to the same resource through the property `ex:partOf`. A more detailed example is provided later in this paper.

Contributions. In order to deal with the previously mentioned issues and solution, we propose the following strategies:

- We propose a strategy for the extraction and *linking* of named entities with KG data based on the integration of Entity Extraction and Linking (EEL) systems (as an ensemble-like strategy). The intuition is that a greater number of coupled extractors provide more extractions than only one single tool.
- We provide a strategy for the *selection* and representation of the named entities contained within a semantic relation. The strategy is based on the closeness of entities to the relation phrase and on the association of named entities with NPs in order to keep the semantic cohesion and coherence[11] of components in a statement.
- We provide a strategy to select and *associate* an identifier for the relation phrase of a semantic relation. For such purpose, the semantic role of the words in the semantic relation is obtained and associated to information provided by a lexical database (with IRI identifiers).
- Based on the previous contribution, we propose a strategy for the representation of elements in a binary semantic relation through a *n*-ary model. For such purpose, we also propose support rules for the selection of entities acting as causer and undergoer of an action in a semantic relation.

Our results demonstrate the benefits of combining Entity Extraction and Linking systems in terms of the F1 measure regarding individual systems. Results also demonstrate that the integration of named entities with grammatical structures provides a better understanding (regarding a baseline system) of proposition-based representations provided by OpenIE for supporting the construction of KGs.

The remainder of this paper is organized as follows. Section 2 provides a background of concepts and related work. Details of the approach are provided in Section 3. Section 4 presents implementation details. Section 5 presents some experiments and results. Finally, conclusions are presented in Section 6.

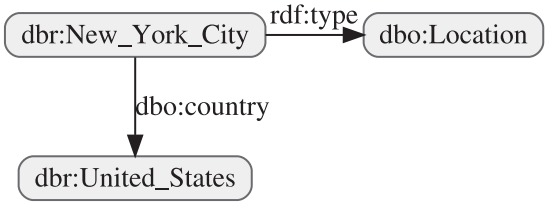

Fig. 2. Example of two connected RDF triples.

2. Background and related work

Before going into details of the proposed approach, we provide general concepts and definitions involved in the KG construction. First, the Semantic Web is presented as a core component for modeling information in a formal fashion. Second, we present two tasks in the Information Extraction area whose purpose is to extract and link named entities and semantic relations from plain text to the Semantic Web. Likewise, we also present related works and discussion.

2.1. The Semantic Web

The Semantic Web (Daconta et al., 2003) is an extension of the traditional Web that allows to formally representing and sharing data through a semantic representation readable by humans and machines. The main component of the Semantic Web is the RDF model,[12] where data is organized with triple elements (*subject, predicate, object*). We formally define an *RDF triple* as follows:

Definition (RDF triple)**.** Given a set of all IRI elements I, a set of blank nodes[13] B, and the set of literals L (datatypes or plain literals), an RDF triple takes the form of $t:=(s, p, o) \in (I \cup B) \times I \times (I \cup B \cup L)$.

Resources in every element of the RDF triple can take different values; the subject may take an IRI or blank node, $s \in (I \cup B)$, the property (predicate) only an IRI resource, $p \in I$, and the object may contain an IRI, blank node or literal value, $o \in (I \cup B \cup L)$.

In the context of the Semantic Web, Färber, Bartscherer, Menne, and Rettinger (2016) refer to Knowledge Graphs (KG) as RDF graphs (graph composed of RDF triples). Along these lines, RDF triple resources can be interlinked to produce a KG, where nodes contain dereferenceable IRIs (for providing additional information of resources through the HTTP protocol) and (directed) edges represent properties taken from existing ontologies from the Semantic Web. Such data organization is possible through the principles provided by the Linked Data initiative[14] We provide an example of a simple RDF graph in Fig. 2, where the resource New York city participates on two RDF triples.

Although there are popular KGs (e.g., DBpedia Lehmann et al., 2015, YAGO Mahdisoltani, Biega, & Suchanek, 2014, and Wikidata Vrandecic & Krötzsch, 2014) composed of millions of RDF triples extracted from (semi) structured data sources like Wikipedia, the automatic construction of KGs from plain text is an

[10] A phrase is a kind of grammatical unit of information that allows for the combination of words into larger units that can act as a sentence element (be part of a subject or an object).

[11] We refer to coherence as a way to envision semantically meaningful units that lead to understanding the meaning of a sentence.

[12] RDF, RDFS, and OWL outline the core components of data representation on the Semantic Web. We only provide basic concepts of such components, for a detailed definition of concepts we refer to Hogan (2013).

[13] Blank nodes are special components used in RDF that by themselves don't identify anything but are useful for grouping other data (acting as parent nodes).

[14] Linked Data (LD) refers to an initiative of the Semantic Web for providing structured information through a model for describing, interlinking and publishing data on the Web.

important and challenging task of the Semantic Web data representation. Given that KGs are composed of RDF triples, the focus of such a task is to identify and associate words within a text to their corresponding elements of an RDF triple. However, this process involves detecting several syntactical and grammatical variations in text (e.g., synonymy, ambiguity) that complicate the automatic interpretation of such elements.

In recent years, Information Extraction (IE) has been a useful area for the extraction and representation of information on the Semantic Web, particularly with two important tasks: named-entity and semantic-relation extraction. We present such tasks in the following subsections.

2.2. Entity Extraction and Linking (EEL)

Entity Extraction and Linking (EEL) refers to the process of extracting named entities from text and linking them to their respective resources from a KG. The process starts with the Named Entity Recognition (NER) task (aka *spotting*), which is in charge of identifying mentions of named entities from text and associating them with a type (e.g., Person, Date, Location, etc.). The second step consists of the Named Entity Disambiguation, whose purpose is to associate such mentions with their corresponding identifiers (IRIs) from a KG (where DBpedia is commonly used).

It is worth mentioning that text documents are composed of *sentences*, and these in turn are composed of grammatical elements such as *nouns* and *verbs*. In this sense, named entities are often represented by *nouns* because these act as names of things. Thus, spotting sometimes takes nouns as mentions in approaches such as AIDA (Hoffart et al., 2011), Babelfy (Moro, Raganato, & Navigli, 2014), among others. However, such mentions (linked to KG instances) do not always represent a complete unit of information as produced by Nominal Phrases (NPs, the represented noun and its modifier). In discourse analysis,[15] NPs play an important role to keep cohesion and coherence of segments in the discourse (Grosz, Joshi, & Weinstein, 1983; Velasco & Rijkhoff, 2008). Therefore, an association that encapsulates entities into NPs is desirable for keeping coherence of ideas in sentences.

Linking entities to resources of a KG is an important principle that facilitates data consumption (Heath & Bizer, 2009). In this regard, EEL systems can be combined in an ensemble-like approach (as presented in Machine Learning Dietterich, 2000) to exploit several features (e.g., domains, KGs, algorithms) in order to compare and obtain better results than using any single such system. Examples of EEL ensemble systems are NERD (Rizzo & Troncy, 2012) and BEL (Zuo, Kasneci, Grütze, & Naumann, 2014).

Novelty. In this paper, we reuse the idea of ensemble systems for extracting and linking entities by means of more than one EEL system. The purpose is to link entities to a KG for the semantic enrichment of text. We provide a strategy to integrate EEL systems and to filter overlapped and duplicated entities. To do so, we rely on a majority-vote strategy as performed by BEL (Zuo et al., 2014) and some filtering criteria. Moreover, in order to preserve coherence at the representation stage, our contribution also relies on the incorporation of a strategy to arrange entities by their corresponding grammatical units of information (NPs).

2.3. Relation Extraction and Linking (REL)

The goal of the Relation Extraction and Linking (REL) task is to represent semantic relations as RDF triples in order to create or populate KGs on the Semantic Web. In a similar fashion to the EEL task, REL is a challenging task composed of sub-tasks for the purpose of extracting and linking entities and semantic relations with resources and properties from a KG and/or ontology. REL involves the following sub-tasks:

1. Entity Extraction (and Linking). The aim of this task is the identification of mentions of entities in text. EEL systems and strategies are usually employed for such a task.
2. Relation parsing. This task identifies semantic relations from text. A semantic relation (Bach & Badaskar, 2007) is a tuple of *arguments* (i.e., entities, things, concepts) with a semantic fragment acting as *predicate* (i.e., noun, verb, preposition). Depending on the number of arguments, a relation may be unary (one argument), binary (two arguments), or n-ary ($n > 2$ arguments).
3. Property selection. This task refers to the alignment of extracted predicates (relation phrases) with a given property from a KG. It is commonly addressed by techniques such as lexical matching (*relation mapping*), training a machine learning algorithm (*distant supervision*), and/or by generating new properties (*property generation*). The selection of an appropriate property guarantee data integrity and interlinking.
4. Representation. According to the kind of information provided by the extracted relations, information can be represented using a single RDF triple (binary relation) or a set of them (n-ary) to express descriptions or additional information (such as provenance) about the data.[16] In any case, such representation requires to fulfill standards of the Semantic Web, which imply the generation and/or selection of identifiers (IRIs) both for entities and for properties.

There are various REL systems that carry out such tasks in some way or another. According to the relation parsing strategy, some REL systems are the following:

Discourse-based. Relations can be extracted by following the idea of Discourse Representation Theory (DRT) (Kamp, Genabith, & Reyle, 2011). DRT is a semantic representation framework for modeling meaning in a logic-based perspective. DRT is based on Discourse Representation Structures (DRS), which allow for representing entities under discussion and information about them in a First-Order-Logic style representation. LODifier (Augenstein, Padó, & Rudolph, 2012) and FRED (Gangemi et al., 2017) are examples of systems using DRT provided by the tool Boxer (Curran, Clark, & Bos, 2007) for extracting relations that are later mapped to KGs and ontologies in an n-ary RDF representation. In the case of FRED, DRSs extracted by Boxer are labeled with the support of Semantic Role Labeling (SRL) (Punyakanok, Roth, & Yih, 2008), whose purpose is to identify predicates, arguments, and their underlying relationship.[17] Finally, FRED represents RDF data with ontologies such as WordNet[18] and DOLCE[19] in order to produce an RDF graph representation. At the top of the FRED approach, Presutti, Nuzzolese, Consoli, Gangemi, and Recupero (2016) propose a method for mapping FRED results to binary relations, where the labels on the path between nodes (identified as potential subject/object pairs) of the graph returned by FRED are concatenated for two options; to match an existing KG property (while not always exists) or to generate a new KG property (with the risk of creating a repeated property), allowing to produce RDF triples.

SRL has also been used as a medium for directly obtaining relations that are later represented in RDF by approaches such as

[15] Discourse refers to a general conceptualization of communication (written or spoken) between two or more people. Thus, discourse analysis studies the use of language defined in terms of coherent sequences of sentences, propositions, speech, and others.

[16] See RDF reification models for more information (Hernández, Hogan, & Krötzsch, 2015).

[17] SRL typically follows the PropBank nomenclature.

[18] WordNet https://wordnet.princeton.edu.

[19] DOLCE ontology http://www.loa.istc.cnr.it/old/DOLCE.html.

Exner and Nugues (2012) and PIKES (Corcoglioniti et al., 2016). Exner and Nugues employ a semantic parser to obtain predicates and arguments that are aligned into RDF relations and finally mapped to DBpedia. PIKES is based on semantic frames for describing events and situations by means of several NLP tasks (e.g., POS, NER, SRL, etc.) and SPARQL-like rules over a KG. The result is finally produced as *n*-ary relations mapped to DBpedia and the FrameBase dataset.[20]

Distant Supervision-based. Another approach for extracting and linking relations is Distant Supervision (DS). It was first proposed by Mintz, Bills, Snow, and Jurafsky (2009), and relies on the hypothesis that, given two entities with a known relation in the KG, sentences in which both entities are mentioned in a text are likely to also mention the relation. DS approaches rely on patterns for obtaining relation mentions and on information extracted from KGs for training machine learning algorithms, whose goal is to classify such relations according to KG facts. Thus, progress on this task is narrowed to machine learning strategies using data from KGs. DS approaches commonly use RDF binary relations to represent data unless some other additional information is added (e.g., contextual data, provenance, descriptions). Examples of DS approaches are Mintz et al. (2009), Augenstein, Maynard, and Ciravegna (2016), and He, Zhang, Hao, Zhang, and Cheng (2017).

OpenIE-based. In order to cover a broad range of relations, Banko, Cafarella, Soderland, Broadhead, and Etzioni (2007) proposed the idea of Open Information Extraction (OpenIE), whose purpose is to extract semantic relations with no restriction about a specific domain. OpenIE has been exploited under implementations that rely on pattern matching and/or machine learning (bootstrap). Approaches such as Liu et al. (2013) and Dutta et al. (2014, 2015) rely on OpenIE systems for extracting relations that are then mapped to the YAGO and DBpedia KGs respectively. Liu et al. (2013) use relations obtained by PATTY (Nakashole, Weikum, & Suchanek, 2013) and then filter them using semantic similarity measures between the relation and properties from the KG. Their final representation consists of binary relations. Dutta et al. (2014, 2015) obtain relations using OpenIE systems (NELL Mitchell et al., 2015 and ReVerb Fader, Soderland, & Etzioni, 2011) that are latter mapped to DBpedia instances and properties. Their strategy relies on mapping relations produced by OpenIE systems to DBpedia using rules and direct associations in order to produce and enrich DBpedia facts.

Novelty. In this paper we propose an REL strategy to generate KGs. In addition to existing systems for parsing and mapping relations to KGs, we propose a method that leverages binary relations produced by OpenIE to further convert them into RDF triples. We opted for such an extraction before other techniques (e.g., DRT, DS) because binary relations represent atomic units of information that convey facts, which can simplify the querying and presentation of data. Thus, we rely on the simplest form of semantic analysis for extracting and representing events in text with no need for producing rules or training models as performed by DS-based approaches that also extract binary relations.

Although the problem of formally representing OpenIE relations has already been discussed in other approaches (Dutta et al., 2014; 2015), we incorporate three novel aspects: a Noun Phrase (NP)-based integration of entities; an SRL-based strategy for obtaining and selecting object causality; and a *n*-ary representation (aka reification). First, and as presented in the previous subsection, the incorporation of an NP-based entity integration preserves the coherence of sentences at the same time that we extract entities from multiple sources. Second, we opted for an OpenIE strategy for relation parsing that has notions of clauses in order to get propositions that express concrete ideas from the input sentence. Nevertheless, it has been demonstrated that SRL can be used to support the OpenIE process (Christensen, Mausam, Soderland, & Etzioni, 2011). Under this premise, we integrate SRL for the detection of predicates, arguments and the roles produced by relation mentions (i.e., causality). Thus, a disambiguation of the sense of the verb and the role of the arguments is obtained by such integration, and later leveraged in the RDF representation. Finally, we follow standards and vocabularies of the Semantic Web for the representation of RDF triples following an *n*-ary assumption. This is because binary relations obtained by an OpenIE tool often include several descriptions and elements that would be impossible to represent them with only one RDF triple.

3. Proposed method

This section presents the proposed method for the construction of Knowledge Graphs from text. As already mentioned, our proposed method is based on a combination of Natural Language Processing (NLP) and Information Extraction (IE) operations in order to transform an input text into RDF triples. In general, such operations involve the acquisition and preprocessing of input text; the extraction of named entities and their association with grammatical units (that help to preserve coherent units of information); the extraction of semantic relations (through an OpenIE approach) and their association with semantic information provided by a Semantic Role Labeling (SRL) approach that lead to identify the *order and selection* of elements to be finally represented through RDF triples. An overview of our method is shown in Fig. 3. Details of every step of the proposed method are provided in the following subsections.

3.1. Document acquisition

This step is intended to perform tasks for collecting and cleaning the text given as input to the method. The proposed method requires plain text[21] as input and thus, several techniques might be involved at this step for extracting and clean it through data parsers. Hence, we consider this step as optional since the source of plain text can be varied (e.g., webpages, documents, etc.) and extracted by different tools. Although this step is inspired by the notion of the Semantic Web to transform the Web into a formal data representation, other text sources may also be accepted as long as the corresponding cleaning and parsing operations are applied. An example of a cleaning operation is depicted in the Listing 1, where an input text with HTML tags is transformed to plain text.

3.2. Preprocessing

Once the plain text is obtained from documents, the next step consists of preprocessing it in order to parse descriptions and elements of information that will be useful for the extraction of named entities and semantic relations. Thus, the following three tasks are applied over the plain text:

- Sentence segmentation. The main idea is to split the input text into sentences. In other words, the text is organized into sequences of small, independent, and grammatically self-contained clauses in preparation for subsequent processing.

[20] FrameBase http://www.framebase.org.

[21] Plain text refers to the data containing readable characters without any graphical object (e.g., images) nor design templates (e.g., tags, tabulations char or line breaks used in webpages).

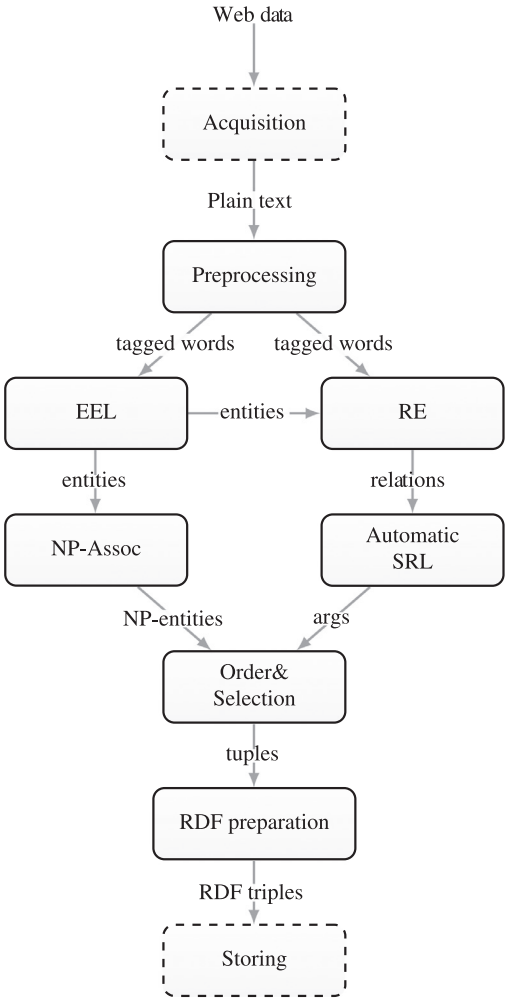

Fig. 3. Overview of the proposed method, where dashed nodes indicate supporting tasks and solid nodes refer to core tasks.

This task is helpful when a text contains several ideas that should be decomposed for a better interpretation.

- Part-Of-Speech (POS). For each word in a sentence, its grammatical category is obtained (e.g., nouns, pronouns, verbs, prepositions, etc). An example of a sentence segmentation and POS tagging is depicted in Listing 2, where two sentences are obtained and tagged with determiners (DT), verbs (VBZ,VBP), Nouns (NN,NNP), among others.
- Syntax tree parsing. Parse trees are required to organize words into groups according to their grammatical sense. We perform a constituency parsing to group words into sub-phrases that function as a single unit. Particularly, we use the obtained *constituency trees* to identify groups of related Nouns denoted by Noun Phrase (NP) units, whose purpose (in our method) is the association of named entities into units of information (later explained in the following subsections). An example of constituency tree is shown in the Fig. 5, where the meaning of some relevant tags is presented in the Table 1.

Table 1
Stanford CoreNLP constituency parser codes.

Tag/Code	Meaning
Root	Text
S	Sentence
NP	Noun Phrase
NNP	Proper noun, singular
NN	Noun, singular or mass
VP	Verb Phrase
VBD	Verb, past tense
CC	Coordinating conjunction

```
Input text:The clinician --Dr. Gregory House--
    ↪ diagnosed a cancer patient in New York
    ↪ City. He is a clinician from New Jersey.
Output text:
1. The_DT clinician_NN --Dr._NNP Gregory_NNP
    ↪ House_NNP-- diagnosed_VBD a_DT cancer_NN
    ↪  patient_NN in_IN New_NNP York_NNP
    ↪ City_NNP.
2. He_PRP is_VBZ a_DT clinician_NN from_IN
    ↪ New_NNP Jersey_NNP.
```

Listing 2. Preprocessing example.

Filtering. In addition to the aforementioned preprocessing tasks, we also include a two-way filtering process in order to exclude sentences that are unlikely to contain entities and relations; thus, less IE operations would be performed while processing a document. Hence, based on ideas stated by Fossati, Dorigatti, and Giuliano (2017), two empirical considerations for filtering sentences are applied:

- Word window. We exclude sentences that are not within a range of word number. We apply the rule $5 > w < 25$ to define that a sentence must contain fewer than 25 words but more than 5. This filtering rule is applied after the sentence segmentation step.
- Syntactic patterns. According to tags provided by the constituency parsing, we filter those sentences described by a basic pattern (e.g., *NP-VB-NP*), where every sentence must contain a Noun Phrase, a Verb, and a Noun Phrase.

3.3. Entity Extraction and Linking (EEL)

In this step, it is performed the extraction and linking of entities to a Knowledge Graph (KG). For such purpose, and with the focus of increasing the number of entities extracted from text, we propose an EEL system based on the idea of ensemble learning systems (as presented in Section 2.2), where the output of various EEL systems is integrated into a single result. However, the difference of our strategy versus ensemble systems centers on the association of entities and grammatical units of information to keep a coherent result (later described in this section). Our proposed strategy to integrate EEL systems considers two aspects: *overlapping entities* (two or more entities sharing the same text fragment) and *duplicate entities* (entities with the same text fragment and IRI). Unless indicated otherwise, throughout this paper we refer to entities

```
Input text:<p>The clinician &ndash;&ndash;<b>Dr.
    ↪ Gregory House</b>&ndash;&ndash; diagnosed a <
    ↪ i>cancer</i> patient in New York City.<p>
Output text:The clinician --Dr. Gregory House--
    ↪ diagnosed a cancer patient in New York City.
```

Listing 1. Cleaning example.

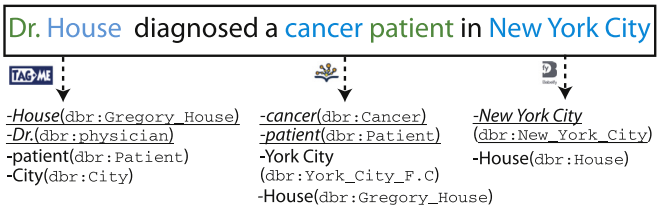

Fig. 4. Example of EEL extractions provided by Tagme, DBpedia Spotlight and Babelfy systems (from left to right respectively). The underlined tuples refer to the tuples selected by the EEL strategy.

as those elements that contain a mention of entity (surface form text) and its identifier from a KG. An example of entities extracted with the proposed method over a fragment of the running example is depicted in the Fig. 4, where three EEL systems are used (*TagMe*[22], *DBpedia Spotlight*[23] and *Babelfy*[24] respectively) and the underlined tuples refer to the final result selected by our strategy. Note that, the running example was shortened for exemplary purposes (*House* instead of *Gregory House*) and the entities only include the mention and the IRI identifier for every entity extracted.

Considering the previous aspects, the EEL strategy proposed in this paper is composed of the following steps:

1. Service invocation. The aim of this step is the invocation of EEL tools and/or services for obtaining mentions of entities and their respective identifiers. We consider public EEL systems (selected in advance) that can be invoked via HTTP requests, where such systems receive as input text the segmented sentences obtained from the preprocessing step, presented in Section 3.2.
2. Output integration. The output of the systems is collected and uniformly transformed (JSON format) considering three aspects: the *surface form* (entity mention), the *IRI* (from a KG), and the *type of entity* (class). However, other criteria may be included as well, such as *offset* (position of words in the sentence), *weighted value* (from disambiguation), and scores from other candidates (contextual data). Duplicated records are allowed at this point for subsequent entity selection decisions.
3. Filtering. The aim of this step is the selection and filtering of entities. Thus, three aspects are considered in this step:
   - Overlaps. Entity overlapping occurs when two or more entities are associated to the same text fragment. For example, in the Fig. 4, three overlaps occur with the text fragment *New York City*; Tagme provides the entity City(dbr:city), DBpedia Spotlight provides York City(dbr:York_City_F.C.) and finally, Babelfy provides New York City (dbr:New_York_City). In this case, we use a natural approach to resolve such overlaps – used by ADEL (Ilievski, Rizzo, van Erp, Plu, & Troncy, 2016) and AGDISTIS (Usbeck et al., 2014) – by expanding entity mentions to a maximal possible match (where the entity New York City is selected).
   - Voting. Inherited from ensemble learning, this aspect covers complete entity overlapping (same mention but different IRI). For example, the mention House(dbr:Gregory_House) is found by the systems *Tagme* and *DBpedia Spotlight*, but *Babelfy* identifies House(dbr:House). The decision is to follow a majority voting scheme where entities recognized for the majority of functions/systems are then selected. In the case of a tie, the first entity is selected and other overlapped entities are discarded. Otherwise, EEL tools can be manually ranked so that the entity returned by the best ranked tool is selected. Although there are other ensemble schemes (Dietterich, 2000), we adopt a majority voting scheme because it is a straightforward technique to implement over collected results.
   - Deduplication. At the end of the voting process, duplicate entities are removed. In other words, entities with the same mention and identifier are removed from the final list to preserve only one.

Additional details of the modules implemented are provided in Section 4.

3.4. NP-association

After we have obtained the entities from the input text, the next step is to associate them with Nominal Phrases (NPs) in order to organize entities into grammatical components (entity cohesion). In other words, this step provides an association of NPs and entities (EEL units) to maintain coherence at representation stage. From the EEL extraction example presented in Fig. 4, we can see that individual entities cancer and patient can be grouped as a single unit of information, which in turn might help with the selection of entities for the construction of RDF triples. The strategy to associate entities with NPs is as follows:

1. NP tagging. The goal of this process is to get the input text labeled with constituent elements (aka. *chunk-tags*) –such as Nominal Phrases (NP), verbs, among others–. This process is usually performed through Natural Language Processing (NLP) tools in two ways: shallow-parsing and constituency parsing. Although both ways enable the extraction of constituent elements, constituency parsing provides better language expressiveness. Thus, the constituency tree obtained in the preprocessing step is used in this task.
2. Entity Extraction. Named entities (and their KG association) are obtained through the EEL step described previously.
3. Entity matching. In this step, entities are grouped into NPs according to a string matching strategy. The surface form (mention) of every entity is compared against the words of the NP and, if they match (partial matchings are accepted), then the entity is selected as *part of* the NP. We mainly consider NPs containing the last leaves of a constituency tree (*individual* NPs). However, there are cases where NPs subsume other NPs, such cases are also considered for the proposed method.
4. Tuple creation. This task produces a list of tuples composed of NPs and their associated entities.

In order to illustrate the NP-association idea, a constituency tree is depicted in Fig. 5, which was obtained by the Stanford CoreNLP tool using the input sentence "*Dr. Gregory House diagnosed a cancer patient in New York City*", where the meaning of some of the used tags is presented in Table 1.[25]

From the entities extracted in Fig. 4 and the constituent tree presented in Fig. 5, the output of the association strategy is presented in Listing 3, where tuples are generated for every individual NP and its involved entities $NP(entity_0, entity_1, ..., entity_{i-1})$. Note that every entity contains an identifier (IRI), but identifiers for NPs are created later at the representation stage. We call *NP-entities* to the final tuples that associate NPs and entities.

The complete NP-association process can be summarized as presented in Algorithm 1.

[22] TagMe https://tagme.d4science.org/tagme/.
[23] DBpedia spotlight http://dbpedia-spotlight.org.
[24] Babelnet http://babelnet.org.

[25] We use POS tags described in the Penn Treebank project https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html.

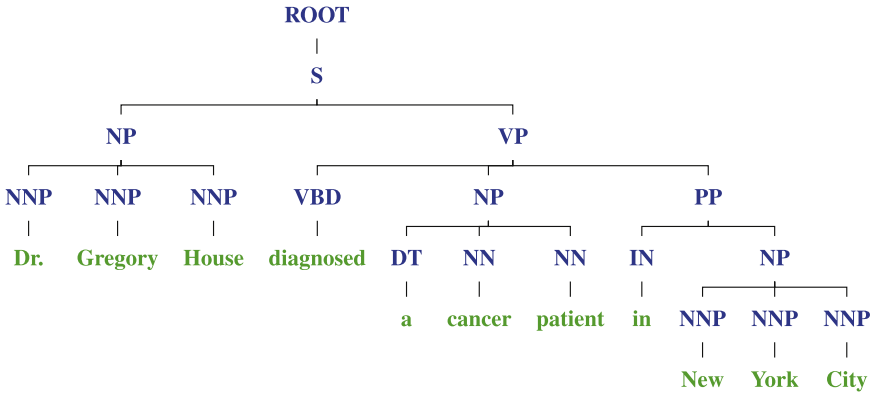

Fig. 5. Constituency tree from the sentence "*Dr. Gregory House diagnosed a cancer patient in New York City*".

```
Input sentence: Dr. Gregory House diagnosed a cancer
    ↪ patient in New York City.
Entities: Dr.(dbr:physician),Gregory House(dbr:
    ↪ Gregory_House), cancer(dbr:Cancer), patient(
    ↪ dbr:Patient), New York City(dbr:New_York_City
    ↪ )

NP-entities: Dr. Gregory House(dbr:physician, dbr:
    ↪ Gregory_House), cancer patient (dbr:Cancer,
    ↪ dbr:Patient), New York City(dbr:New_York_City
    ↪ )
```

Listing 3. Association of entities with NPs.

Algorithm 1: Association of entities with NP tags.

Data: PLAINTEXT SENTENCE, EEL ENTITIES
Result: *NP_entities*
1 CHUNK-TAGS ← OBTAINCONSTITUENCY(SENTENCE)*NP_entities* ← {∅};
2 *NPs* ← FILTERNPS(CHUNK-TAGS); /* Keep NP chunks only $\{np_0, np_1, \ldots, np_{j-1}\}$ */
3 **forall the** $np \in NPs$ **do**
4 *assocEntities* ← {∅};
5 **forall the** $ne \in EEL$ **do** /* Iterate over entities */
6 **if** $ne.SF \subseteq np$ **then** /* Matching surface form (SF) against NP */
7 *assocEntities* ← *assocEntities* ∪ *ne*;
8 **end**
9 **end**
10 *NP_entities.append*(⟨*np*, *assocEntities*⟩);
11 **end**

3.5. Relation Extraction (RE)

Once information about named entities has been obtained, the next step is to determine the semantic relation between these named entities, in such a way that new facts about the world are extracted. Hence, we consider those relations involving *actions* where real world things are described as true (propositions). The decision of extracting binary relations is related to representing different kinds of information; for example, definitions/inclusion (`New York is a city`) or facts/case relations (e.g., `the dog eats kibbles`). The former (definitions) usually represent hyponymy relations that are commonly tackled with lexical patterns, such as proposed by Hearst (1992) and Snow, Jurafsky, and Ng (2004). However, fact relations (case relationships) are complex to understand and represent because there are several kinds of expressions and verb tenses that may be combined to express several ideas about a single referenced object. In this regard, the OpenIE approach (Banko et al., 2007) is capable of dealing with complex sentences. Hence, this step has the purpose of obtaining semantic relations from plain text using an OpenIE tool.

3.6. Automatic SRL

Even though semantic relations extracted by OpenIE approaches have the form Predicate(Subject, Object), the sense of the predicate (tense, conjugation) can change the interpretation of subject and object in a relation to denote the causer of an action and the modified object. Thus, the structure of the relation might be reorganized for the construction of RDF triples.

Along these lines, *frame semantics* is a linguistic theory that allows for obtaining linguistic meaning through related concepts in terms of *predicates* and *arguments* (Fillmore, 1976). In other words, frame semantics has the purpose of identifying arguments in a sentence and their specific thematic roles that help to indicate the causer of an action and the modified object (the undergoer of an action) using *Semantic Role Labeling* (SRL) (Gangemi, 2013) –as employed by approaches such as FRED (Gangemi et al., 2017) and PIKES (Corcoglioniti et al., 2016). Hence, elements in a sentence conveying causality are considered as the *Agent* role (commonly identified with the `A0` code) and those elements receiving an action refer to the *Patient* role (commonly identified with the `A1` code).

Two traditional lexical resources used to annotate a corpus with semantic information of words are PropBank[26] and FrameNet.[27] Hence, in this step, we identify thematic roles in a semantic relation in order to determine the position (in subject or object) of entities as causer or undergoer of an action/event. That is to say, the causer may appear on the Subject or Object of a sematic relation (the same for the undergoer). To illustrate this case, an example of Semantic Role Labeling[28] is shown in Listing 4, where two semantically similar input sentences in active voice and passive voice (*Sentence1* and *Sentence2*, respectively) provide different relations but the same predicate sense and arguments (`A0` is identified in the object of *Sentence2* but in the subject of *Sentence1* and the other way round for `A1`). The sense of the predicate is a key feature for the property selection presented in the following Section. Note that the NP-entity associated to the roles is included for demonstrative purposes.

PropBank tags are commonly used by SRL for assigning the thematic role (aka. Verb Modifiers) for the arguments. Some popular

[26] PropBank https://verbs.colorado.edu/~mpalmer/projects/ace.html.
[27] FrameNet https://framenet.icsi.berkeley.edu/fndrupal/.
[28] SRL examples were processed through the web service http://cogcomp.org/page/demo_view/srl.

```
Sentence1: Dr. Gregory House diagnosed a cancer
    ↪ patient in New York City.
Relation: [Dr. Gregory House, diagnosed, a cancer
    ↪ patient in New York City]

Sentence2: A cancer patient was diagnosed by Dr.
    ↪ House in New York City.
Relation: [A cancer patient, was diagnosed, by Dr.
    ↪ House in New York City]

Same output:
Predicate: diagnose.01
A0: Dr.; NP-entity: Dr. Gregory House(dbr:Physician,
    ↪ dbr:Gregory_House)
A1: patient ; NP-entity: a cancer patient(dbr:Cancer
    ↪ , dbr:Patient)
LOC: New York City ; NP-entity: New York City (dbr:
    ↪ New_York_City)
```

Listing 4. Example of Semantic Role Labeling.

Table 2
Popular PropBank thematic roles.

MOD	Declaration
A0	Agent/causer
A1	Patient/undergoer
A2	Starting point/attribute
TMP	When some action takes place
LOC	Where some action takes place
NEG	Negation
CAU	Reason for an action
ADV	Adverbial

such roles (MOD) are presented in Table 2. The A0, A1 and LOC roles are within the focus of this work.

3.7. Order and selection

Information extracted up to this stage has been transformed from unstructured (plain text) to a (semi-) structured fashion, identifying components such as named entities, grammatical associations, semantic relations, and roles of arguments. Thus, the next step is to put all these pieces together into a formal representation following the Linked Data principles and standards. Hence, this step provides the necessary components to create RDF triples in terms of the order and selection of resources (named entities) and descriptions (properties) clearly identified with IRIs.

3.7.1. Entity selection

The goal of this step is to associate NP-entities with the predicate arguments (thematic roles) identified in a semantic relation. In other words, we use SRL to identify predicates, their correct and disambiguated sense, and their arguments from a sentence in order to discover the role of entities involved in the subject and object of a semantic relation (extracted from the same sentence). This is performed through a matching process between NP-entities, SRL arguments, and elements from a semantic relation. Nevertheless, the matching process is not simple because SRL tools might not recognize some arguments in a sentence. Hence, our approach considers some intuitions for selecting and assigning roles to entities that belong to a relation.

The ideal case is to correctly obtain arguments for the *Agent* and *Patient* of a predicate described by A0 and A1 respectively (as presented in the example of Listing 4). However, there could be a varied number of situations regarding the identification of arguments by SRL approaches. Hence, according to the thematic roles identified by SRL tools, we propose a strategy that considers three common cases for the selection of entities that are subsumed in a semantic relation:

1. Correct identification. This case occurs when the arguments and roles of a predicate are completely identified from a sentence and finally associated with entities. This is the simplest case, and the selection of entities is given by a string comparison between surface forms of entities and arguments of the predicate. However, given the previous association of entities with NPs (NP-entities), the output consists of NP-entities and their role with respect to elements in the semantic relation. A particular case considered as correct identification is given by the identification of higher roles. For example, when there is no A0 role, but there are A1 and A2 roles; in this case, the *Agent* is given by A1 and the *Patient* by A2 because the first is performing as a logical subject and it is assumed to be the one who did something.
2. Partial identification. For this case, only one argument (and its role) from the sentence is identified. Thus, the idea is to perform a search process in order to identify the element (subject or object) in the semantic relation that contains the identified argument. For example, if A0 (representing the *Agent*) is identified and associated with entities derived from the subject of the semantic relation, then it is assumed that the *Patient* (A1) should be obtained from entities in the object. We assume that entities close to the verb in a semantic relation are the most important participants in an event. As such, the unrecognized argument is obtained through this principle.
3. No identification. This situation occurs when none of the roles were identified. On the premise that a semantic relation describes something about the subject, and the object refers to what is said about the subject, it is assumed that the subject performs the role of *Agent* and the object as *Patient*. The entities are thus obtained through the previously introduced strategy (case 2) to select the closest entities to the verb.

Although there are other roles that indicate modification of predicates, for practical reasons we only consider *Agent* and *Patient* roles. For example, MOD-LOC[29] and MOD-TMP[30] are only considered as participants of an event (LOC is included as additional data in Listing 4).

The process for the association and selection of entities regarding predicate arguments is summarized in the Algorithm 2 . Although the SRL process can return many predicates (and senses), the only one considered is that matching the verb of the semantic relation (line 1). Arguments correctly identified (case 1) are compared against the NP-entities to get the corresponding resources (line 5). In the case of partial argument identification (case 2), the identified argument is processed as *Agent* or *Patient* accordingly. Subsequently, we identify the position of the semantic relation (subject or object) whose entities were not taken from the processed argument (line 11) in order to select the missing argument assumed to be close to the verb (line 12). Finally, for unknown arguments (case 3), the closest entities to the verb in the subject and object of the semantic relation are assumed to be the *Agent* and *Patient* respectively (lines 19 and 20).

3.7.2. Property selection

The output of the previous step provides entities and their thematic roles, which are capable of being part of resources in an RDF triple. However, the property is an essential element in the triple that still need to be identified. This step presents a strategy for the selection of a property identifier that best describes a semantic relation phrase. Our strategy relies on retrieving the IRI for predicate senses obtained by an SRL tool.

[29] MOD-LOC is a semantic role which expresses a location argument in a sentence.
[30] MOD-TMP indicates the moment when some action takes place.

Algorithm 2: Association between predicate arguments and NP-entities.

Data: NP_ENTITIES:ENT, SRLPREDICATES, RE:R
Result: NP_ENTITIES selected for representation

```
pred ← MATCHING(R.PREDICATE,SRLPREDICATES);   /* Get SRL predicate matching the verb phrase in the Semantic Relation */
case ← DETERMINECASE(pred);   /* Every predicate has arguments A0, A1, AN */
if case = 1 then   /* Complete identification of arguments */
    /* Search for an NP-entity matching any of the arguments */
    agent ← ENTITYMATCHING(pred.arg0,ENT);
    patient ← ENTITYMATCHING(pred.arg1,ENT);
end
if case = 2 then /* Partial identification */
    if pred.arg0 then   /* If argument identified is 0 */
        agent ← ENTITYMATCHING(pred.arg0,ENT);
        /* Identifies if entities are taken from subject or object of the semantic relation */ position ← GETRELPOSITION(agent);
        patient ← GETNEARESTENTITY(ENT.position);
    else
        start assigning patient;
        /* The same process starting with patient */
    end
end
if case = 3 then   /* No identification */
    agent ← GETNEARESTENTITY(ENT.subject);
    patient ← GETNEARESTENTITY(ENT.object);
end
```

```
SELECT ?predIdentifier WHERE
{
        ?predIdentifier rdfs:label $predicateForm .
}
```

Listing 5. SPARQL query used for obtaining the resource linked to a predicate sense.

According to the example presented in Listing 4, the SRL detected the predicate sense (diagnose.01) according to the context and tense of the input sentence. In other words, the correct predicate sense is disambiguated from a set of possible senses obtained from the PropBank lexical database. For our proposed method, such predicate sense describes the predicate (relation phrase) element of the semantic relation. Once the predicate sense is known, an important aspect is to associate it to an identifier from a KG. In this regard, the Premon[31] KG provides resources associated to SRL predicate models and arguments –such as PropBank, NomBank among others–, where every resource is identified by an IRI. Thus, we must obtain the resource of the KG that contains the same predicate sense as the one obtained by the SRL for the processed sentence. As we noted, the predicate sense is assigned as literal value to resources in Premon through the rdfs:label property.

Therefore, a quick procedure for obtaining the identifier is through a SPARQL[32] query submitted to the endpoint[33] of Premon. We propose the SPARQL query shown in Listing 5, where the variable $predicateForm contains the predicate sense. Thus, the idea is to find the resource whose label is denoted by that sense (theoretically unique).

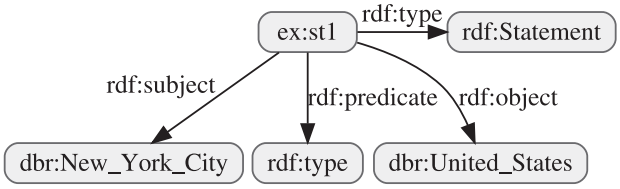

Fig. 6. Example of standard RDF reification.

3.8. RDF preparation

After all main components are extracted (NP-entities, semantic relations, SRL annotations), the final step is to put all data together for creating RDF triples in a persistent format. RDF provides a model based on binary relations but sometimes a relation needs to be modeled involving several resources and descriptions. This is the case of *n*-ary relations, where two or more resources are involved for describing an RDF statement. As such, RDF statements with several descriptions follow a model called *reification* (Fossati et al., 2017), in which every element of a triple should be individually specified through RDF triples (whose result is an RDF graph). An example of a standard reification is depicted in Fig. 6, where the statement *New York City is a Location* takes four RDF triples to be represented.

The standard reification is often criticized because of its lack of conciseness to represent an RDF statement and due to the difficulty to query data (Berners-Lee, 2010). Thus, instead of representing triples using the standard reification, we rather prefer a *n*-ary relation representation (Hernández et al., 2015), where a resource is involved in a relationship that may contain diverse qualifiers and participants (Dodds & Davis, 2011). Finally, once entities, relations, arguments (roles), and predicate identifiers are obtained, the results are integrated to produce RDF triples organized into a *named graph*, where a set of RDF triples can be considered to be named by an identifier (in the case of documents or triples from a webpage).

In order to better demonstrate the final output of our proposed method, an extended version of the running example is shown in Listing 6. In this case, we present some additional information. First, the declaration of the used RDF vocabulary namespaces at the top of the Listing (e.g., gold: is an ontology for the linguistics domain used in the declaration of the *Agent* and *Patient*). Second, elements of the main *Event* are also declared such as the *predicate sense*, the *Agent*, and the *Patient*. Third, we also include the NP-entities and the entities that compose them (with the property dcterms:isPartOf). Finally, the original input sentence and the semantic relation from which the RDF triples were extracted are also represented (assigned with the nif vocabulary and the rdf:comment property respectively). Note that we use the cvst vocabulary as a local context to define the IRI event (e.g., cvst:d1evt5), the named graph (based on the assumed provenance webpage cvst:Dr_Greg_House.html), and NP-entities (e.g., cvst:cancer_patient). Along these lines, the format used for the serialization of the triples is TriG,[34] which is composed of RDF triples and named graphs in a reduced text format. It is worth mentioning that, for this example, we represented only one event/relationship but many others can be represented from different sentences within the same original webpage processed.

[31] Premon http://premon.fbk.eu.
[32] SPARQL is an SQL-like language used to query and present information contained in a Knowledge Graph.
[33] https://premon.fbk.eu/query.html.
[34] TriG format https://www.w3.org/TR/trig/.

```
@prefix gold:  <http://purl.org/linguistics/gold/> .
@prefix pmn:   <http://premon.fbk.eu/resource/> .
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix dbr:   <http://dbpedia.org/resource/> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix cvst:  <http://www.tamps.cinvestav.mx/> .
@prefix nif:   <http://persistence.uni-leipzig.org
      ↪ /nlp2rdf/ontologies/nif-core#> .
@prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#> .
@prefix event: <http://purl.org/NET/c4dm/event.owl#> .

cvst:Dr_Greg_House.html {

cvst:d1evt5 a pmn:pb215-diagnose.01 , event:Event ;
  gold:agent    cvst:Dr_Gregory_House ;
  gold:patient  cvst:cancer_patient ;
  rdfs:comment "diagnosed(Dr. Gregory House, a cancer patient in
       ↪ New York City)" ;
  nif:referenceContext cvst:Dr_Greg_House.html#sentence1 .

dbr:Gregory_House dcterms:isPartOf cvst:Dr_Gregory_House .
dbr:Physician dcterms:isPartOf cvst:Dr_Gregory_House .
dbr:Cancer dcterms:isPartOf cvst:cancer_patient .
dbr:Patient dcterms:isPartOf cvst:cancer_patient .

cvst:Dr_Greg_House.html#sentence1 a nif:Sentence, nif:Context ;
  nif:isString "The clinician ––Dr. Gregory House–– diagnosed a
       ↪ cancer patient in New York City" .


}
```

Listing 6. RDF representation example.

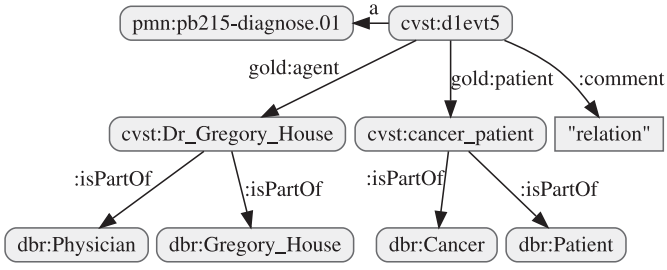

Fig. 7. *n*-ary RDF representation example from a binary relation.

A visual representation of the example presented in Listing 6 is shown in the Fig. 7, where the main elements were represented by an RDF graph. In this case, we first create a node that acts as the main event (`cvst:d1evt5`), where the type of such a resource is the predicate sense obtained by the SRL step and queried against the Premon KG. Descriptions for the event are added following the arguments of the relation. Note that values for *Patient* and *Agent* come from an NP-entity and thus, identifiers (IRIs) were minted and associated to its respective subsumed entities. For space reasons, we do not draw aspects of the named graph, the original sentence, and some name spaces are omitted.

Optionally, the output produced can be stored in an RDF Graph store supporting named graphs (e.g., OpenLink Virtuoso Erling, 2012). This step is optional since some users may prefer to keep data as RDF files.

The strategy for the selection of properties and storage is depicted in the Algorithm 3, where proposition (line 1) obtains the arguments and roles, and propertyMapping (line 2) obtains an identifier for the predicate by submitting a SPARQL query with the predicate sense returned by SRL. Extracted elements are represented into RDF triples (line 3), and finally the result is saved into an RDF store (line 4).

4. Implementation details

We implemented our proposal for constructing KGs as a Java application. Some internal configuration details of the Information Extraction and NLP tools and services applied by such application are provided in this Section with respect to the architecture depicted in Fig. 3.

Algorithm 3: Property selection.

Data: NP_ENTITIES:ENT, SRLPREDICATES, REL:R

1 *proposition* ← GETPROPOSITION(ENT, SRLPREDICATES, R); `/* Get Agent and Patient resources from Algorithm 2 */`

2 *propertyMapping* ← SUBMITQUERY(*proposition.predicate*); `/* query predicate to get identifier */`

3 *RDF facts* ← REPRESENT(R, *proposition*, *propertyMapping*); `/* For representation */`

4 STORE(*RDF facts*);

Acquisition. The Web is a vast repository of information containing noisy and low-quality data. However, as a quality consideration, the input text of the proposed method must be properly written in terms of correct and coherent ideas. Thus, as an initial consideration, we use documents extracted from webpages provided by IT news. Along these lines, for our proposal we detected that RSS providers represent a wealth data source for extracting IT news webpages. Although data might already be obtained by users, we performed the following strategy for obtaining a sample of documents from the Web using RSS providers.

- Document acquisition: Nine RSS providers were manually obtained from different online IT news websites such as BBC,[35] DailyTech,[36] and Computer-Weekly.[37] Subsequently, we captured the links inside those providers (using a script) for further processing. A total of 605 webpage links were collected.
- Download: Webpages were downloaded using a HTTP client.[38] Some of the sites required cookie based identification or a web browser user agent.
- Content extraction: Webpages were processed by the Jericho parser[39] for cleaning and obtaining the plain text.

A total of 605 documents were downloaded through the proposed strategy, where a total of 12,015 segmented sentences were extracted (after the preprocessing step described below). It is worth mentioning that as a proof of concept we apply our proposed method over Information Technologies (IT) news webpages. In principle, we use such a domain because it is a special interest topic for academic and business purposes (Elliott, 2017; Foote & Halawi, 2018). Additionally, we also highlight two aspects regarding the data source: availability of contents (through the Internet) and well-written documents (with complete and unambiguous ideas). The former aspect is in accordance with one of the original purposes of the Semantic Web to formally represent contents of the Web. The latter regarding the quality of texts required by the proposed method to process ideas with presumable less writing issues (as provided by news webpages) than other data sources such as social networks and web blogs (Salloum, Al-Emran, Monem, & Shaalan, 2017). However, we should mention that more domains and types of documents (not only webpage contents) are within the interest of our future work.

Preprocessing. The NLP tasks considered at the Preprocessing step were performed through the Stanford CoreNLP (Manning et al., 2014) tool; using data models for English.[40] Such tasks involve the

[35] BBC news http://www.bbc.com.
[36] DailyTech http://www.dailytech.com.
[37] ComputerWeekly http://www.computerweekly.com.
[38] Apache HTTP client https://hc.apache.org.
[39] Jericho parser http://jericho.htmlparser.net/docs/index.html.
[40] Stanford CoreNLP models https://stanfordnlp.github.io/CoreNLP/.

tokenization of words, allowing the *sentence segmentation*, the *Part of Speech* (POS) tagging, and the structural parsing (*constituency tree*). Additionally, we also performed a strategy to expand language contractions, for example, converting the word *aren't* into *are not*.

EEL. We propose the integration of existing Entity Extraction and Linking systems in order to obtain a high number of entities from the input text than using any single such system. In this sense, we integrate the output provided by distinct EEL systems available through web services. Thus, the services were invoked by means of an HTTP client and POST requests (EEL systems such as Babelfy and TagMe require a registration to get an access key to use the service). In general, the EEL systems typically require some input parameters (to be stated within the HTTP request) such as *confidence, type of extraction, input text, language, output format*, among others. As introduced in the example of Fig. 4, three EEL systems were selected for extracting and linking entities: DBpedia Spotlight, Babelfy, and TagMe. For all three such systems we use as input parameters the segmented sentences as *input text* (one for request), we set entities linked to a KG as *type of extraction* (systems such as Babelfy may only return entities without a link to an external KG), English as the *language* of the input text, and JSON as *output format* returned by the extractors. The *confidence value* (aka. *support*) refers to the degree to which a mention is linked to a KG resource as the most likely match. This value is defined as a threshold for limiting the number of entities extracted from text (increasing the precision at the cost of a probably lower recall). The confidence values used in our implementation are DBpedia Spotlight (0.35), Babelfy (0.01), and TagMe (0.06). Settings of such parameters were defined by experimental studies such as (Usbeck et al., 2015) and by experiments that provide a balance between number of extractions and precision. Note that the name of the variables used as input parameter varies for every system and then, it is recommended to check their available APIs for further details. For example, the variable of the confidence in DBpedia Spotlight is expressed as *confidence* but as *th* in Babelfy.

Of course, many other systems can be invoked and integrated into the final output but we only use such tools for simplicity (as they have public APIs) and based on performance results reported in the literature (Usbeck et al., 2015).

NP-association. No additional configurations were required for this module. However, we used a JSON parser[41] in order to process the output provided by the EEL systems.

RE. Through an initial evaluation performed over some OpenIE tools (presented later in Section 5), ClausIE (Del Corro & Gemulla, 2013) reported the best precision with respect to two other OpenIE tools. Hence, for extracting relations, we selected and configured ClausIE using default parameters to obtain only binary relations. Moreover, the CC option was set as active in order to couple entities connected by *coordinated conjunctions*.[42] Note that the output required by this step consists of binary relations provided by OpenIE tools and thus, other strategies can replace the output provided by ClausIE.

Automatic SRL. The library Mate-Tools[43] was used for obtaining semantic roles associated to NP-entities and predicates of semantic relations. Thus, predicates and arguments provided by Mate-Tools are based on annotations of the lexical resource PropBank. The data models used by Mate-Tools for internally parsing, lemmatizing and tagging were the CoNLL2009 models for English.[44] Mate-Tools was used according to results reported in the literature (Roth & Woodsend, 2014). However, other lexical resources (FrameNet, VerbNet) and tools can be incorporated to determine thematic roles as presented in Giuglea and Moschitti (2006, 2004).

Order & selection. Particular implementations for this step are presented for the property selection. In order to obtain an identifier for the event/action expressed in the semantic relation, we leverage the predicate sense identified by Mate-tools to perform a SPARQL query (presented in Listing 5) over the Premon KG as previously explained in Section 3.7.2. For such purpose, we implemented a Jena[45] module (using the SPARQL 1.1 syntax) with the query and the endpoint already mentioned.

RDF preparation. Information obtained throughout the pipeline of our proposal is then represented on RDF triples. For such purpose, we developed a Jena module for organizing all event-based information obtained from sentences and documents. In other words, we represent events, where an event consists of a predicate and its arguments (*Agent* and *Patient*), which are represented by an *n*-ary reification model using the TriG format. We use TriG because it allows to define compact and readable RDF statements that can be organized within named graphs as presented in the example of Listing 6.

5. Evaluation

Evaluation of REL-based approaches is not a straightforward task because there are no standard criteria for assessing a particular data representation output. Most times, human intervention is required to verify the quality of represented data due to the lack of gold standard datasets. Hence, in this paper, we follow an *a posteriori* assessment of the output, where the process starts by processing plain text to obtain entities, extract relations, represent RDF, and the output is finally verified by human judges. In this way, we evaluate our approach regarding three main components: entities, relations, and RDF representation. All the tests were carried out on a computer with 8GB RAM, Processor Intel core i5 (2.7 GHz), and OS X Yosemite. Details of the evaluation are provided in the following subsections.

5.1. Dataset

The information used for the experiments was retrieved with the strategy presented in the Acquisition step, described in the Section 4. For demonstrating the performance of our proposal, a sample of 605 IT news webpages were downloaded through such strategy. Then, we evaluated the EEL and RE steps with 100 randomly selected sentences from such data. Finally, the representation was evaluated using the complete set of downloaded webpages.

5.2. Metrics

For such an evaluation, standard IE metrics (e.g., precision, recall, F–measure) were applied and, in the case of multiple human judges, a Kappa-based metric for measuring the inter-rater agreement was applied. These measures are obtained as follows:

[41] https://github.com/fangyidong/json-simple .

[42] A coordinated conjunction (CC) is a conjunction that connects two or more parts of the sentence (e.g., *and, but, or*, among others.

[43] MatePlus https://github.com/microth/mateplus.

[44] Data models downloaded from https://code.google.com/archive/p/mate-tools/downloads.

[45] Jena https://jena.apache.org.

Precision. Precision (P) specifies the correct amount of information retrieved. In other words, it refers to the proportion of correct members assigned to a class that are really members of that class. It can be obtained by the formula (1).

$$P = \frac{\texttt{Correct Elements Obtained}}{\texttt{Total Elements Obtained}} \tag{1}$$

Recall. Recall (R) represents the degree of correct information retrieved. In other words, it is the proportion of class members that the system assigns to the class. It can be obtained by the formula (2).

$$R = \frac{\texttt{Correct Elements Obtained}}{\texttt{Total Elements Correct}} \tag{2}$$

F–measure. F-measure is used for combining the values of precision and recall in one metric as presented in formula (3).

$$F = \frac{(B^2+1)PR}{B^2P+R} \tag{3}$$

From the formula (3), precision and recall get an equal importance when B is equal to one, in such case, the metric is called the harmonic mean ($F1$).

Agreement. Taking into account the number of human judges, an overall agreement needs to be obtained.[46] As such, the number of human judges and categories are considered for scoring an item. However, agreement may occur by "chance" and thus, for avoiding a biased result the agreement among judges needs to be obtained. Typically, the Kappa (Randolph, 2005) value is used to determine the inter-rater agreement between observed data and prior data, which is described in the formula (4).

$$K_{free} = \frac{\left[\frac{1}{Nn(n-1)}\left(\sum_{i=1}^{N}\sum_{j=1}^{k} n_{ij^2} - Nn\right)\right] - \left[\frac{1}{k}\right]}{1 - \left[\frac{1}{k}\right]} \tag{4}$$

where N is the number of cases, n is the number of human judges, k is the number of categories, and n_{ij} is the number of judges who assigned the case to the same category; see Randolph (2005) for a complete example.

5.3. EEL evaluation

Purpose. This evaluation has the purpose of verifying the effectiveness of some EEL systems applied to a sample of documents downloaded by the strategy presented in Section 4. Through this evaluation, we assess the accuracy produced by the output of some EEL systems and the output produced by our proposed integration strategy.

Scenario. For practical reasons, we tested some EEL systems that provide an API through a publicly available web service and have top results as reported in Usbeck et al. (2015). Thus, Babelfy, DBpedia Spotlight, and TagMe services were tested under parameters reported in Section 4. For such purposes, 100 randomly selected sentences from the retrieved IT news documents were used as input for such systems. As previously described, the testing data consists of IT news extracted from webpages. In the case of the selected sentences, a human judge with knowledge about such domain was in charge of evaluating the output produced by the EEL systems. In consequence, the testing sample has in average 24.27 words and 614 named entities linked to DBpedia. Finally, the metrics *precision* (P), *recall* (R), and *F1* were used to evaluate the final result, where a successful extraction case is when both a mention of an entity and its identifier from a KG are correctly identified (exact matching).

Table 3
EEL evaluation.

System/metric	P	R	F1	# Ent.
Babelfy	0.6419	0.6074	0.6242	581
Spotlight	**0.8560**	0.5325	0.6566	382
Tagme	0.7787	0.8897	0.8305	705
Our system	0.8139	**0.9643**	**0.8827**	731

Results. Results of this evaluation are presented in Table 3, where the systems Babelfy, DBpedia Spotlight (Spotlight), and Tagme are compared against our EEL integration strategy. Note that the last column indicates the total number of entities extracted by every system.

Discussion. The results presented in Table 3 indicate the accuracy of some popular EEL systems regarding plain text obtained from IT news webpages. We compared the results provided by individual systems against the one provided by our strategy for integrating the output of all three such systems into one output. These results demonstrate that our strategy provides better performance in terms of the F1 measure. Although the DBpedia Spotlight system obtained better precision on the evaluation, it also obtained the lowest number of entities and thus, the lowest recall. Likewise, in some cases, false positive entities given by systems negatively impact the final result of our integration strategy.

5.4. RE evaluation

Purpose. Since the proposed method relies on relations extracted by an OpenIE system, we want to see the effectiveness of some such systems with respect to data retrieved from web documents, so that the output of an OpenIE system can be integrated into our representation approach.

Scenario. Several OpenIE tools have been developed since the approach described by Banko et al. (2007). However, some of them are succeeded in favor of new systems that solve existing issues. For such reasons, and according to the tools described by Zouaq, Gagnon, and Jean-Louis (2017), we selected and evaluated three recent OpenIE systems: Stanford OIE[47], Open Information Extraction 4 (OIE4)[48], and ClausIE.[49] For the evaluation, we downloaded and locally installed such tools and configured them under default parameters for obtaining binary relations. In the case of the input text, a sample of 100 sentences was randomly selected from the IT news documents.

Results. The output of the selected tools was evaluated by a human annotator. The results are shown in Table 4, where the used metrics are *precision* (P), *recall* (R), and *F1*.

Discussion. Although OIE4 provided the best recall in the evaluation, it also obtained many incorrect extractions and thus, a low precision was obtained. Given the results provided in this evaluation and those provided by Emani, Silva, Fiés, and Ghodous (2016),

[46] The overall agreement refers to the relative amount of samples in which the human evaluators agree for a particular feature (Van den Berge, Schouten, Boomstra, van Drunen Littel, & Braakman, 1979).

[47] Stanford OpenIE http://nlp.stanford.edu/software/openie.shtml.

[48] OIE tool http://openie.allenai.org/.

[49] ClausIE https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/software/clausie/.

Table 4
RE evaluation.

System/Metric	P	R	F1
Stanford OIE	0.255	0.467	0.330
OIE4	0.461	**0.782**	0.580
ClausIE	**0.638**	0.620	**0.628**

and Del Corro and Gemulla (2013), ClausIE was the best tool. Hence, our RDF representation strategy applies ClausIE for obtaining relations between entities.

5.5. RDF representation evaluation

Taking as input the complete dataset, entities and semantic relations extracted by EEL and RE systems respectively were combined to create RDF triples according to the standards and formats of the Semantic Web. Hence, to assess the effectiveness of the information represented through the proposed approach, we performed two evaluations regarding the number of triples represented (quantitative) and the precision of such data (qualitative).

5.5.1. Quantitative analysis

Purpose. The goal of this experiment is to analyze our strategy in terms of the number of RDF triples generated.

Scenario. As previously stated, the proposed strategy was implemented in Java. Details of the implementation are provided in Sections 3 and 4. Thus, for this evaluation, a total of 12,015 sentences from the IT news documents were processed for constructing RDF statements.

Results. The results collected in this evaluation are presented in Table 5, where **EEL** refers to the extracted and linked entities, **RE** refers to extracted relations, **RE-represented** refers to the number of events (extracted relations) which are finally represented in RDF, and **RDF statements** refers to the total number of triples (counting associations of NP-entities and events).

Discussion. An interesting fact presented in Table 5 is regarding the number of relations (RE) finally represented as RDF, which is less than a third of the total number of identified relations. This fact is produced because only those relations containing resources (named entities) in *subject* and *object* are represented in RDF (e.g., dbr:New_York_City rdf:type dbr:Location. Although literal values can be assigned to the object of an RDF triple (e.g., dbr:New_York_City rdfs:label "New York"), such cases are not within the focus of this work.

5.5.2. Qualitative analysis

Purpose. After representing RDF triples, the following step was to evaluate the quality of the represented data.

Scenario. As previously mentioned, evaluation of relations in the Semantic Web is not a straightforward task because there are no available gold-standard datasets for comparison. The choice of most approaches is to apply a manual evaluation scheme, where a determined number of triples is taken and reviewed by a determined number of human judges. However, to the best of our knowledge, there are no standard criteria defined for such a manual evaluation: the attributes and number of human evaluators, the features to evaluate (e.g., domain, input sentences), the accuracy and clarity of the relation elements (Subject-Relation-Object), and the coherence or final interpretation of the RDF statement (appropriate IRIs and properties) (Dutta et al., 2015; Freitas, Carvalho, Da Silva, O'Riain, & Curry, 2012).

Description	Content	Correct	Incorrect
Sentence	The clinician -Dr. Gregory House- diagnosed a cancer patient in New York City		
Semantic Relation	**Subject** Dr. Gregory House **Predicate** diagnosed **Object** a cancer patient in New York City	◉	○
Agent	http://www.tamps.cinvestav.mx/Dr_Gregory_House |-http://dbpedia.org/resource/Physician |-http://dbpedia.org/resource/Gregory_House	○	○
Action	http://premon.fbk.eu/resource/pb215-diagnose.01 definition: give	○	○
Patient	http://www.tamps.cinvestav.mx/cancer_patient |-http://dbpedia.org/resource/Cancer |-http://dbpedia.org/resource/Patient	○	○

Fig. 8. Evaluation form presented to judges where two options allow to indicate if an element is correct or not.

Hence, we perform an experiment based on the strategy proposed by Dutta et al. (2015), where a set of triples is presented to a human judge for evaluation, in which every element of the triple needs to be marked as correct (including the semantic relation) to deem the entire statement as precise. Hence, we developed a web application where a set of events were presented to human judges under some criteria. First, evaluation was performed by 4 students (judges) from an IT-based engineering college. The system processed English sentences, thus, the events extracted are represented as well. Therefore, the required judges have, at least, an intermediate level of English (e.g., to read and understand IT news in English). Likewise, judges have notions of the terminology and structure used on the RDF representation (e.g., RDF triples, thematic roles). Second, every judge evaluated 50 events (consisting of four triples each) to decide if these were correct or not in terms of *Agent*, predicate, *Patient*, and the semantic relation. An example of the web form is depicted in Fig. 8, where the main elements of the event to be evaluated are the semantic relations, *Agent*, predicate sense, *Patient*, and the involved NP-entities. Note that the original input sentence is included to help users take decisions and understand ideas.

Results. Given the different evaluations provided by the four judges, the precision values are depicted in Table 6, where individual values were obtained per element and with respect to all judges. Note that we refer to the precision of subject and object given the Agent and Patient respectively because it is supposed that such roles define the element described and its value.

Agreement among evaluators. In this case, a free-marginal multirater Kappa (Randolph, 2005) was obtained giving as criteria 4 judges (raters), 50 subjects (cases), and 2 categories.[50] The result of the agreement measure is provided in Table 7.

Baseline. Finally, we provide a comparative of the results obtained by our complete proposal against a version of our same system without the association of Noun Phrases and entities (NP-entities). The purpose is to demonstrate the benefit of associating NPs and entities regarding the coherence of the final representation. In this case, the same 50 events described at the scenario of this subsection were used for testing. The results of the experiment are depicted in Table 8, where the *complete system* refers to our approach

[50] Kappa implementation https://gist.github.com/ShinNoNoir/9687179.

Table 5
RDF stats.

	Sentences	EEL	RE	RE-represented	RDF statements
Total	12,015	103,401	41,190	12,686	89,486
Average/Doc	20.6089	177.3602	70.6518	21.7598	153.4922

Table 6
Precision of represented data. Precision of Subject (PS), Precision of Predicate (PP), Precision of Object (PO), Precision of Relation (PR), and Precision of Triples (PT).

User	PS	PP	PO	PR	PT
1	0.72	0.88	0.70	0.78	0.52
2	0.72	0.90	0.58	0.88	0.50
3	0.58	0.78	0.50	0.80	0.46
4	0.74	0.94	0.76	0.84	0.64
Median	0.72	0.89	0.64	0.82	0.51

Table 7
Inter-rater agreement.

Evaluation	Overall agreement	Kappa
Subject	0.7266	0.4533
Predicate	0.8233	0.6466
Object	0.6566	0.3133
Relation	0.8500	0.7000
Triple	0.6900	0.3800

Table 8
Comparison of the whole approach against a baseline version (without association of NPs and entities). Precision of Subject (PS), Precision of Predicate (PP), Precision of Object (PO), Precision of Relation (PR), and Precision of Triples (PT).

System	PS	PP	PO	PR	PT
Complete system	0.72	0.89	0.64	0.82	0.51
Baseline	0.50	0.90	0.46	0.80	0.26

with the association of NP-entities (median values are taken from Table 6).

These results indicate that our approach based on the association of NPs and entities helps to preserve the coherence at the representation of RDF triples. Note that the precision of the relation and the predicate sees no significant change since the strategy does not affect the output provided by the OpenIE and the predicate sense given by SRL.

Discussion. Our experiments indicate encouraging results in terms of understanding by human judges. Moreover, although there are approaches such as Dutta et al. (2015); Fossati et al. (2017); Nebhi (2013) and Presutti et al. (2016) that present a precision greater than 75%, authors pursue a controlled test (with a small number of triples, defined number of classes, and a specific domain). This is due to two main reasons: first, the evaluation of a large number of instances is very expensive or unfeasible in terms of time and effort required by human judges; second, a fixed domain and classes facilitate the selection of human judges participating in the evaluation. Thus, this fact may lead to obtain higher precision values of more than 90% in some approaches that manually map relations to properties using predefined domain-specific heuristics and rules (Dutta et al., 2015; Nebhi, 2013).

On the other hand, the obtained overall agreement among human judges is around 0.65 to 0.85, which means that judges share decisions most of the times. Thus, the Kappa value seems to be from *fair* to *moderate* according to the interpretation of Kappa given by Landis and Koch (1977). Hence, human evaluators depict a fair to moderate agreement that the represented data is precise and coherent, which demonstrates that the evaluation was not guided by chance. In terms of Kappa values from other approaches, Relext (Schutz & Buitelaar, 2005) obtained 0.27 and FrameBase (Rouces, de Melo, & Hose, 2015) 0.23 with a precision of 0.24 and 0.78, respectively. In general, this fact is observed because the agreement by human judges is mainly guided by their subjectivity such that even high-level experts can take different decisions. However, other aspects influencing such agreement are:

- Text-based. This aspect comprises the clarity of text in which problems of interpretation may be observed. Complex sentences are included in this aspect, in which more than two ideas can be included in a same sentence. Moreover, according to the text, sentences may be discarded if are very formal or containing technicality, which in principle should not be a problem for a human domain expert. Finally, the amount of text presented to the evaluator may obfuscate their decisions (e.g., number of cases, categories or options, text length, etc.).
- System-based. The final extraction of a system such as the proposed in this research work is produced by NLP tools and strategies. However, the accuracy of such tools may not be exact and thus, the final result may affect the data interpretation by judges. For example, extracting an element incorrectly may encourage people to reject the entire case. Moreover, evaluators may get confused if extractions are correct but not belong to the core idea of a complex sentence.
- Human-based. Some errors are product of the judges because of an unclear or unfamiliar understanding of the reviewed concepts by them.

Despite these issues, the results presented in Table 8 demonstrate that our strategy to associate NPs and entities improves the coherence of RDF represented facts.

6. Conclusions

Most of the information consumed by human users on the Web has an unstructured nature, which makes it difficult to be processed by applications unless complex tasks are performed. To address this issue, the Semantic Web provides a way to structure all the information through data models, standards, vocabularies, and tools. Although this seems to be a solution to improving information consumption, representing information on a formal structure is a very complex and time-consuming process because unstructured data do not have features and descriptions to support a formal representation.

This paper proposed an approach for constructing Knowledge Graphs on the Semantic Web through a task we coined as Relation Extraction and Linking. Our approach relies on Information Extraction (IE) tasks for obtaining named entities and relations to then link them using data and standards of the Semantic Web. Moreover, we integrated information from such IE tasks together with grammatical units of information for keeping coherence at the representation stage. For such purposes, we propose two important components:

- The output of EEL tools (i.e., named entities with IRI identifiers) can be associated with semantic information provided by Noun

Phrases (NP) in order to keep coherent statements. Our intuition is that NPs represent grammatical units of information and thus, can be used to organize entities where the original idea is preserved.
- The combination of results from traditional IE tools and IE tools incorporating semantic web data has been helpful for obtaining semantic relations and named entities respectively. Moreover, the complete result together with semantic information of thematic roles provided by Semantic Role Labeling and Knowledge Graphs allow producing events in the form of RDF statements.

Although our solution covers relations useful for specific representations of sentences, users may face contrasting data needs that are better addressed by other kinds of approaches for a comprehensive linguistic analysis, or representations based on Semantic Web data.

Future work. We plan to employ the information represented by our strategy to support applications in areas such as information retrieval, information extraction, and question answering. We will also intend to leverage such information as intermediate structures for the construction of direct (non-reified) binary relations. Finally, we plan to evaluate our proposed method with different domains and types of documents (not only webpage contents).

Acknowledgment

We would like to thank Aidan Hogan for its comments on this paper.

References

Augenstein, I., Maynard, D., & Ciravegna, F. (2016). Distantly supervised web relation extraction for knowledge base population. *Semantic Web Journal, 7*(4), 335–349.
Augenstein, I., Padó, S., & Rudolph, S. (2012). LODifier: Generating linked data from unstructured text. In *Extended semantic web conference* (pp. 210–224). Springer.
Bach, N., & Badaskar, S. (2007). A review of relation extraction. *Literature review for language and statistics II*.
Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., & Etzioni, O. (2007). Open information extraction from the web. In *International joint conferences on artificial intelligence (IJCAI)* (pp. 2670–2676).
Van den Berge, J., Schouten, H., Boomstra, S., van Drunen Littel, S., & Braakman, R. (1979). Interobserver agreement in assessment of ocular signs in coma. *Journal of Neurology, Neurosurgery & Psychiatry, 42*(12), 1163–1168.
Berners-Lee, T. (2010). The future of rdf. https://www.w3.org/DesignIssues/RDF-Future.html. [Online; accessed January-09-2018].
Christensen, J., Mausam, Soderland, S., & Etzioni, O. (2011). An analysis of open information extraction based on semantic role labeling. In *International conference on knowledge capture* (pp. 113–120). ACM.
Corcoglioniti, F., Rospocher, M., & Aprosio, A. P. (2016). Frame-based ontology population with PIKES. *IEEE Transactions on Knowledge and Data Engineering, 28*(12), 3261–3275.
Curran, J. R., Clark, S., & Bos, J. (2007). Linguistically motivated large-scale NLP with c&c and Boxer. *Annual meeting of the association for computational linguistics*. ACL.
Daconta, M. C., Obrst, L. J., & Smith, K. T. (2003). *The semantic web: A guide to the future of xml, web services, and knowledge management*. Wiley Publishing, Inc.
Del Corro, L., & Gemulla, R. (2013). Clausie: Clause-based open information extraction. In *International conference on world wide web* (pp. 355–366). ACM.
Dietterich, T. G. (2000). Ensemble methods in machine learning. In *International workshop on multiple classifier systems* (pp. 1–15). Springer.
Dodds, L., & Davis, I. (2011). In L. Dodds, & I. Davis (Eds.), *Linked data patterns: A pattern catalogue for modelling, publishing, and consuming linked data*.
Dutta, A., Meilicke, C., & Stuckenschmidt, H. (2014). Semantifying triples from open information extraction systems. *Frontiers in Artificial Intelligence and Applications, 264*, 111–120.
Dutta, A., Meilicke, C., & Stuckenschmidt, H. (2015). Enriching structured knowledge with open information. In *International conference on world wide web* (pp. 267–277). WWW Steering Committee.
Ehrlinger, L., & Wöß, W. (2016). Towards a definition of knowledge graphs. *Semantics-pd*. CEUR-WS.org.
Elliott, S. W. (2017). Projecting the impact of information technology on work and skills in the 2030s. *The Oxford handbook of skills and training*.
Emani, C. K., Silva, C. F. D., Fiés, B., & Ghodous, P. (2016). Improving open information extraction for semantic web tasks. *Transactions on Computational Collective Intelligence, 21*, 139–158.
Erling, O. (2012). Virtuoso, a hybrid rdbms/graph column store. *IEEE Data Engineering Bulletin, 35*, 3–8.
Exner, P., & Nugues, P. (2012). Entity extraction: From unstructured text to dbpedia rdf triples. *The web of linked entities workshop*. CEUR.
Fader, A., Soderland, S., & Etzioni, O. (2011). Identifying relations for open information extraction. In *Conference on empirical methods in natural language processing* (pp. 1535–1545). ACL.
Färber, M., Bartscherer, F., Menne, C., & Rettinger, A. (2016). Linked data quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO. *Semantic Web, 9*(1), 1–53.
Fillmore, C. J. (1976). Frame semantics and the nature of language. *Annals of the New York Academy of Sciences, 280*(1), 20–32.
Foote, A., & Halawi, L. A. (2018). Knowledge management models within information technology projects. *Journal of Computer Information Systems, 58*(1), 89–97.
Fossati, M., Dorigatti, E., & Giuliano, C. (2017). N-ary relation extraction for simultaneous t-box and a-box knowledge base augmentation. *Semantic Web Journal*.
Freitas, A., Carvalho, D. S., Da Silva, J. C., O'Riain, S., & Curry, E. (2012). A semantic best-effort approach for extracting structured discourse graphs from wikipedia. *Workshop on the web of linked entities at ISWC*.
Gangemi, A. (2013). A comparison of knowledge extraction tools for the semantic web. In *Extended semantic web conference* (pp. 351–366). Springer.
Gangemi, A., Presutti, V., Recupero, D. R., Nuzzolese, A. G., Draicchio, F., & Mongiovì, M. (2017). Semantic web machine reading with FRED. *Semantic Web, 8*(6), 873–893.
Giuglea, A., & Moschitti, A. (2006). Semantic role labeling via framenet, verbnet and propbank. In *Computational linguistics and annual meeting of the ACL* (pp. 929–936). ACL.
Giuglea, A.-M., & Moschitti, A. (2004). Knowledge discovering using framenet, verbnet and propbank. *Workshop on ontology and knowledge discovering at ECML*.
Grosz, B. J., Joshi, A. K., & Weinstein, S. (1983). Providing a unified account of definite noun phrases in discourse. In *Annual meeting on association for computational linguistics* (pp. 44–50). ACL.
He, D., Zhang, H., Hao, W., Zhang, R., & Cheng, K. (2017). A customized attention-based long short-term memory network for distant supervised relation extraction. *Neural Computation, 29*(7), 1964–1985.
Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. In *Conference on computational linguistics* (pp. 539–545). ACL.
Heath, T., & Bizer, C. (2009). *Evolving the web into a global data space*. Reading, Massachusetts: Morgan and Claypool publishers.
Hernández, D., Hogan, A., & Krötzsch, M. (2015). Reifying RDF: What works well with wikidata? In *International workshop on scalable semantic web knowledge base systems* (pp. 32–47). CEUR-WS.org.
Hoffart, J., Yosef, M. A., Bordino, I., Fürstenau, H., Pinkal, M., Spaniol, M., et al. (2011). Robust disambiguation of named entities in text. In *Empirical methods in natural language processing* (pp. 782–792). Association for Computational Linguistics.
Hogan, A. (2013). Linked data and the semantic web standards. *Linked data and the semantic web standards*. Chapman and Hall/CRC Press.
Ilievski, F., Rizzo, G., van Erp, M., Plu, J., & Troncy, R. (2016). Context-enhanced adaptive entity linking. *International conference on language resources and evaluation*. European Language Resources Association (ELRA).
Kamp, H., Genabith, J. V., & Reyle, U. (2011). *Discourse representation theory* (15, pp. 125–394)). Springer.
Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 159–174.
Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., et al. (2015). DBpedia – A large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web, 6*(2), 167–195.
Liu, F., He, S., Liu, S., Zhou, G., Liu, K., & Zhao, J. (2013). Open relation mapping based on instances and semantics expansion. In *Asia information retrieval societies conference* (pp. 320–331). Springer.
Mahdisoltani, F., Biega, J., & Suchanek, F. M. (2014). Yago3: A knowledge base from multilingual wikipedias. *Biennial conference on innovative data systems research*. CIDR.
Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Annual meeting of the association for computational linguistics* (pp. 55–60). The Association for Computer Linguistics.
Mintz, M., Bills, S., Snow, R., & Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Annual meeting of the ACL and natural language processing of the AFNLP* (pp. 1003–1011). ACL.
Mitchell, T., Cohen, W., Hruschka, E., Talukdar, P., Betteridge, J., Carlson, A., et al. (2015). Never-ending learning. In *Conference on artificial intelligence (AAAI)* (pp. 2302–2310). AAAI Press.
Moro, A., Raganato, A., & Navigli, R. (2014). Entity linking meets word sense disambiguation: A unified approach. *Transactions of the Association for Computational Linguistics, 2*, 231–244.
Nakashole, N., Weikum, G., & Suchanek, F. M. (2013). Discovering semantic relations from the web and organizing them with PATTY. *SIGMOD Record, 42*(2), 29–34.
Nebhi, K. (2013). A rule-based relation extraction system using DBpedia and syntactic parsing. In *International conference on NLP and DBpedia: 1064* (pp. 74–79). CEUR-WS.org.
Presutti, V., Nuzzolese, A. G., Consoli, S., Gangemi, A., & Recupero, D. R. (2016). From hyperlinks to semantic web properties using open knowledge extraction. *Semantic Web, 7*(4), 351–378.
Punyakanok, V., Roth, D., & Yih, W. (2008). The importance of syntactic parsing and inference in semantic role labeling. *Computational Linguistics, 34*(2), 257–287.
Randolph, J. J. (2005). Free-marginal multirater kappa (multirater k [free]): An alternative to fleiss' fixed-marginal multirater kappa. *Joensuu Learning and Instruction Symposium*.

Rizzo, G., & Troncy, R. (2012). NERD: A framework for unifying named entity recognition and disambiguation extraction tools. In *European chapter of the ACL* (pp. 73–76). ACL.

Roth, M., & Woodsend, K. (2014). Composition of word representations improves semantic role labelling. In *Empirical methods on natural language processing.* (pp. 407–413). ACL.

Rouces, J., de Melo, G., & Hose, K. (2015). Framebase: Representing n-ary relations using semantic frames. In *European semantic web conference* (pp. 505–521). Springer.

Salloum, S. A., Al-Emran, M., Monem, A. A., & Shaalan, K. (2017). A survey of text mining in social media: Facebook and twitter perspectives. *Advances in Science, Technology and Engineering Systems Journal*.

Schutz, A., & Buitelaar, P. (2005). Relext: A tool for relation extraction from text in ontology extension. In *International semantic web conference* (pp. 593–606). Springer.

Snow, R., Jurafsky, D., & Ng, A. Y. (2004). Learning syntactic patterns for automatic hypernym discovery. *Advances in Neural Information Processing Systems*, 1297–1304.

Tanwar, M., Duggal, R., & Khatri, S. K. (2015). Unravelling unstructured data: A wealth of information in big data. In *Reliability, infocom technologies and optimization* (pp. 1–6). IEEE.

Usbeck, R., Ngomo, A.-C. N., Röder, M., Gerber, D., Coelho, S. A., Auer, S., et al. (2014). AGDISTIS – Graph-based disambiguation of named entities using linked data. In *International semantic web conference* (pp. 457–471). Springer.

Usbeck, R., Röder, M., Ngonga Ngomo, A.-C., Baron, C., Both, A., Brümmer, M., et al. (2015). GERBIL: General entity annotator benchmarking framework. In *World wide web conference* (pp. 1133–1143). ACM.

Velasco, D. G., & Rijkhoff, J. (2008). *The noun phrase in functional discourse grammar*: 195. Walter de Gruyter.

Vrandecic, D., & Krötzsch, M. (2014). Wikidata: A free collaborative knowledgebase. *Communications of the ACM, 57*(10), 78–85. doi:10.1145/2629489.

Zouaq, A., Gagnon, M., & Jean-Louis, L. (2017). An assessment of open relation extraction systems for the semantic web. *Information System, 71*, 228–239.

Zuo, Z., Kasneci, G., Grütze, T., & Naumann, F. (2014). BEL: Bagging for entity linking. In *International conference on computational linguistics* (pp. 2075–2086). ACL.