Journal Title
XX(X):1–23

Reprints and permission:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

Mining information from sentences through Semantic Web data and information extraction tasks

Jose L. Martinez-Rodriguez[1,3] and Ivan Lopez-Arevalo[1] and Ana B. Rios-Alvarado[2]

Abstract

The Semantic Web provides guidelines for the representation of information about real-world objects (entities) and their relations (properties). This is helpful for the dissemination and consumption of information by people and applications. However, the information is mainly contained within natural language sentences, which do not have a structure or linguistic descriptions ready to be directly processed by computers. Thus, the challenge is to identify and extract the elements of information that can be represented. Hence, this paper presents a strategy to extract information from sentences and its representation with Semantic Web standards. Our strategy involves Information Extraction tasks and a hybrid semantic similarity measure to get entities and relations that are later associated with individuals and properties from a Knowledge Base to create RDF triples (Subject-Predicate-Object structures). The experiments demonstrate the feasibility of our method and that it outperforms the accuracy provided by a pattern-based method from the literature.

Keywords

Semantic Web, Information Representation, Information Extraction, RDF representation, Linked Data

1 Introduction

Extracting information from text represents an activity that implies the identification of (relevant) objects and their connections, guiding the declaration of descriptions or ideas about things. Such information can

[1] Cinvestav-Tamaulipas, Victoria, Mexico
[2] Faculty of Engineering and Science, Autonomous University of Tamaulipas, Victoria, Mexico
[3] Multidisciplinary Academic Unit Reynosa-Rodhe, Autonomous University of Tamaulipas, Reynosa, Mexico

Corresponding author:
Jose L. Martinez-Rodriguez, Computer Systems Laboratory, Multidisciplinary Academic Unit Reynosa-Rodhe, Autonomous University of Tamaulipas, Reynosa, Tamaulipas, Mexico.
Email: lazaro.martinez@uat.edu.mx

be useful in a variety of tasks related to areas such as Information Retrieval, Question Answering, and Data Mining, to mention a few.

Nowadays, data sources such as the Web allow people to browse an immeasurable number of documents containing information that can be exploited; regularly in the form of sentences with statements that contain a verb and arguments (e.g., *The sun is a star*) that can be expressed with RDF triples. However, due to the large scale of such a data source, handling the extraction of information represents an unfeasible task for humans. Moreover, in case that a computer supports this process, the challenge is to parse information elements because the sentences do not show explicit linguistic descriptions or a structure able to be directly processed by computers.

In recent years, two areas involved in the extraction and representation of information are the Information Extraction (IE) and the Semantic Web. First, IE is aimed at obtaining structured data from an unstructured data source, where two tasks are highlighted: named entity recognition and semantic relation extraction. While the former task refers to the identification of real-world objects (e.g., names of People, Places, Companies) the latter refers to the relation between such entities (often in form of actions). Second, the Semantic Web provides guidelines and standards for the representation of information, which defines the RDF triple (Subject-Predicate-Object) as the basic unit of information. In this sense, entities can be part of Subject and Object within an RDF triple, which are described by a relation defined as the Predicate (obtained from a vocabulary or ontology).

Hence, the difficulty of identifying information from sentences and representing it as RDF triples relies on the extraction of named entities and their semantic relations. However, in the Semantic Web, elements in such components must be associated with resources (URIs) from a Knowledge Base (KB), such that, unique identifiers are assigned to them to facilitate the identification, description, and usage of the information. For example, the sentence *"Leo Messi plays as a forward for the Barcelona soccer team"*, contains the semantic relations `Play as(Leo Messi, forward)` and `Play for(Leo Messi, Barcelona soccer team)`, where an RDF representation is shown in Figure 1. The combination of triples can be organized in the form of a graph, where nodes represent entities/resources and edges refer to predicates/properties. For readability purposes, we use fictitious names for resources from the Wikidata KB [1]; for example, wd:Leo_Messi (wd:Q615), wd:fcBarcelona (wd:Q10467), wdt:playsFor (wdt:P54), and so on, where `wd` and `wdt` refers to URI prefixes (later defined).

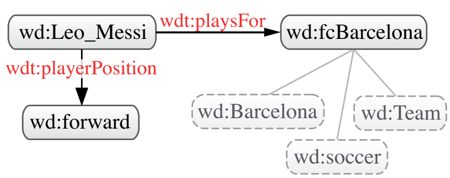

Figure 1. Example of RDF representation. Dashed nodes indicate hypothetical entities.

The extraction of entities from text and their linking with resources from a KB is a task known as Entity Extraction and Linking (EEL), which has been widely studied in the literature [2, 3, 4, 5]. Note that, in Figure 1, we also included (blurred) entities (wd:Barcelona, wd:soccer, wd:team) that can be recognized instead of wd:fcBarcelona according to the used EEL tool and its training models (results may vary). On the other hand, the subsequent joining of the entities found in a text through predicates (from a semantic relation) linked with properties from a KB refers to a task known as Relation Extraction and Linking (REL), which is the main focus of this work. Such a task has been addressed in diverse ways that provide advantages and difficulties. For example, by a strategy known as Distant Supervision [6], a machine learning algorithm is trained with existing KB data for the recognition (classification) of new RDF triples. However, it requires to select the training data and to configure the parameters of the selected algorithms. Another strategy is to find the relation between entities (through patterns regularly [7]) and then mapping it to a property of a KB in a task known as *property selection*. The difficulty of this strategy is the generation of the mapping rules and the limitation of properties to link only specific types of properties (e.g., properties between People and Companies). Additionally, both approaches do not often indicate the process of entity selection. That is, in the hypothetical case that the three entities (Barcelona, soccer, team) are identified instead of wd:fcBarcelona (Figure 1), which one should be selected for the representation of the RDF triple?

This paper proposes a method for the representation of information by extracting RDF triples from natural language sentences. The proposed method is based on a combination of Information Extraction tasks for obtaining entities, semantic relations, and their linking with resources from a KB. First, entities are obtained through an existing strategy for the integration of public EEL services. Second, semantic relations among entities are obtained through an OpenIE strategy that is not guided by a domain of information. Finally, we propose a property selection strategy for mapping the predicate of the semantic relation with its respective property from a KB. This is based on a hybrid semantic similarity measure that compares the predicate against a set of property candidates derived from the KB to select the more similar one. Unlike existing approaches, we do not rely on training data or specific pattern mappings for the generation of the RDF triples.

In summary, the contributions of this work are as follows:

- A set of recommendations for the selection of named entities and semantic relations likely to be represented as RDF triples.
- The proposal of a property selection strategy based on a hybrid sentence similarity measure and information from Wikidata.
- A strategy for the automatic extraction of RDF triples from unstructured text in English to produce a knowledge graph.
- A prototype, which integrates NLP tools for implementing a version of the proposed strategy.
- Guidelines for the evaluation of the prototype by judges, managing distinct granularities to determine the accuracy of RDF triples.

The remainder of this paper is organized as follows: the related works are presented in Section 2. Section 3 presents our representation strategy. Details of an implementation of the representation strategy are presented in Section 4. In Section 5, the experiments are described. Finally, the conclusions are presented in Section 6.

2 Related Work

The representation of information as RDF triples generally involves the recognition and linking of named entities, finding their association through the relation extraction task, and then linking the predicate with a property from a KB. Since there are diverse EEL approaches used for extracting entities in a sentence [2, 5, 8], we organize this section according to those works extracting binary relations and further parsing them into RDF triples. We identified two kinds of approaches: feature-based and pattern-based.

2.1 Feature-based

Semantic and syntactic features can be used to obtain semantic relations from text. The purpose is to label a text (with manual intervention or NLP tools) to obtain features used in a machine learning algorithm, mainly a supervised strategy. However, such labeling is a time-consuming task, where instances for training and testing must be prepared/labeled by human experts. This aspect has been addressed by two strategies: 1) the incorporation of *crowdsourcing* tasks and 2) the use of existing KB data.

Regarding the first strategy, Fossati *et al.* [9] trained a supervised classifier supported by Lexical Units (LU). Every LU is composed of POS tagging describing elements such as verbs and their arguments. LUs are ranked with TF-IDF and standard deviation measures in order to select the top-N meaningful LUs. Subsequently, a training set is labeled through a crowdsourcing platform, which is used to extract facts by means of a supervised strategy (SVM and a baseline); considering as categories Frame Elements (event descriptions and their participants) derived from the FrameNet lexical resource.

On the other side, Distantly Supervised (DS) approaches rely on large KBs such as DBpedia or YAGO for training a classifier following the idea "*if two entities participate in a relation and both entities are contained in a sentence then it expresses the same relation*". Diverse DS approaches have been proposed to leverage KBs available as Linked Data such as YAGO used by Nguyen [10] and Freebase, used by Augenstein [11]. The common strategies used for classification by DS approaches are multiclass logistic regression [11], neural networks [12], and deep learning [13], to mention a few. Regarding the property linking, DS approaches associate patterns and features (extracted from textual relation mentions) with a particular KB property. That is, properties extracted from a KB are used within the training process to directly associate a particular property. Although this approach provides high levels of accuracy, the difficulty lies in preparing the training data, which involves the selection of triples from the KB and, in diverse cases, such data only cover specific types of entities and properties.

2.2 Pattern-based

Approaches in this category are aimed at generating patterns (pattern induction) that describe general relations from the text. For example, Exner and Nugues [14], DeepDive [15] and Nguyen and Moschitti [10] used a dependency parser in order to extract a syntactic structure from relation mentions. However, diverse approaches reuse the idea proposed by Banko et al. [16] for extracting OpenIE relations, those derived through lexical patterns from the dependency tree of a sentence (with no restriction of a domain of information) and where the extracted relations help to obtain new ones. Approaches such as Dutta *et al.* [7], Liu *et al.* [17], and Soderland and Mandhani [18] are within this setting. Note that both strategies, based on dependency parsing, obtain relations that will later be linked with an ontology or KB.

Regarding the property linking, Dutta *et al.* [19, 7] mapped the relations extracted by OpenIE systems to DBpedia properties. The process involves a mapping of entities (within the parsed relation) to the KB using SPARQL queries so that an existing property between the two entities is filtered by rules or directly selected. Another strategy is to generate the property from the predicate of the relation phrase. The concatenation of the relation phrase words generates the property label to thereafter include some definitions or descriptions about the property (e.g. domain and range obtained from the entities).

Legalo [20] generates predicate labels by creating OWL properties using the *CamelCase* notation together with information (from the entities) such as the domain, range, linguistic evidence (original sentence), among others. Likewise, RDFLiveNews [21] produces a descriptive label for relation clusters sharing the same meaning, this label is mapped to `owl:ObjectProperty` considering the domain and range of the subject and object types.

The advantage of using OpenIE by these approaches is to get general relations not attached to a single domain. However, the property linking process is often limited to specific types of entities and direct mappings that require manual intervention by a human expert. Additionally, the generation of properties relies on the identification of entities to provide details and restrictions for the generated property, which in some cases may produce inconsistencies and/or duplicated items. Thus, to avoid such limitations, we provide an information representation strategy that leverages EEL systems and an OpenIE approach to obtain entities and their relation. The property is selected by matching the predicate from the relation with property candidates (obtained from a KB) by means of semantic similarity measures.

3 Representation strategy

The aim of this work is to represent RDF triples extracted from sentences using data and standards of the Semantic Web. The proposed strategy is depicted in Figure 2.

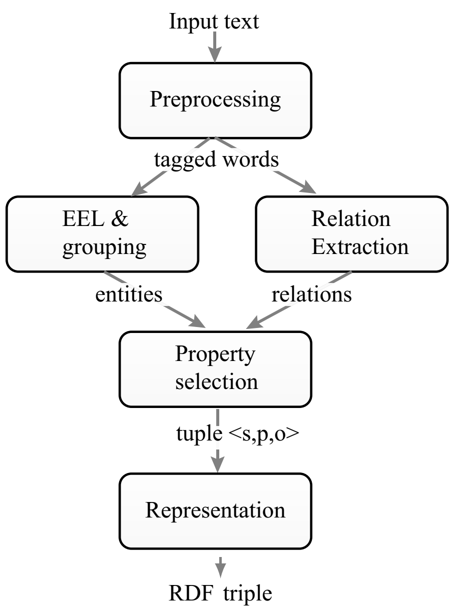

Figure 2. Overview of the proposed strategy.

The general idea is that given a number of entities extracted from a semantic relation (from the subject and object), a set of property candidates is collected from a KB, where the property that best represents the relation between such entities is selected through a disambiguation process (considering the semantic similarity of the predicate with the labels of the property candidates). Additional details are provided in the following subsections.

3.1 Preprocessing

This stage performs cleaning and parsing tasks to obtain features used in further steps of the extraction and representation of entities and relations. The applied preprocessing tasks are sentence segmentation, Part of Speech (POS) tagging, and dependency tree parsing.

3.2 EEL & grouping

Entities are extracted at this stage through the use of EEL systems. However, this stage goes one step further, by applying an organization of the found entities in such a way that they keep together as a single element able to define a coherent unit of information. That is, more than one entity can appear in the subject or object of a semantic relation, but if we separate them, the original idea of the sentence might be missed. In this sense, we perform the same strategy presented by Martinez-Rodriguez et al. [22], which consists of the following steps:

- Entities are first recognized by a combination of publicly available EEL systems integrated in an ensemble-like setting, where such systems are invoked via HTTP requests and later filtered to avoid duplicated and/or overlapped items.
- The found entities are organized according to the Nominal Phrase (NP) the entity belongs to. NPs (provided by the dependency parsing) are grammatical units of information that allow words to be combined into larger units that can act as a single sentence element. Thus, we create an entity for the NP (called NP-entity), which contains n entities (in a kind of composition relation).

3.3 Relation Extraction

After recognizing entities through the previous stage, this stage obtains binary relations (between entities) through lexical patterns derived from the dependency tree of a sentence with no restriction of a domain of information as provided by the OpenIE approach [16].

3.4 Property selection

This stage associates the predicate of a semantic relation with a property from a KB. The involved steps are presented in the following subsections.

3.4.1 Property collector. The purpose of this step is to collect a set of property candidates (from a KB) that represent possible descriptions (connections) between the entities extracted from a semantic relation. The proposed strategy is based on capturing the result of a SPARQL query executed over a KB endpoint. This step involves two phases: query preparation and submission/parsing.

First, the aim of the query preparation is to construct a query that returns the property candidates between two (or more) entities. The basic idea is depicted in Figure 3, in which we look for existing properties linking two "seed" entities extracted from the subject and object of a semantic relation. The strategy is to query those existing properties (`?p`) between the subject (`:subject`) and object (`:object`) entities and additionally between the individuals of their same class (siblings obtained from

`?classSub` and `?classObj`). The idea is to obtain as much as possible the properties related to the input entities. However, a Cartesian product (`Sx?pxO`) is produced among (likely thousands of) individuals in the KB, which may degrade the performance. Thus, the number of siblings could be restricted when the computational resources are limited.

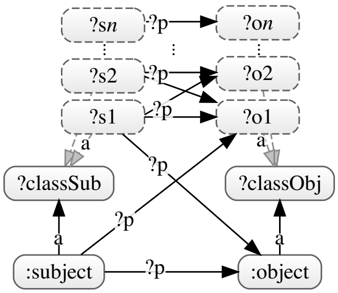

Figure 3. Properties between subject and object and between their siblings are collected.

Second, in the submission/parsing phase, the generated query is executed in a SPARQL endpoint and the resulting properties are retrieved.

3.4.2 Disambiguator. This step is aimed at selecting the property that best describes the input semantic relation. Thus, the proposed strategy is based on a measure that determines the similarity between the predicate of a semantic relation and each property from the set of candidates, where the most similar property (the one with the higher score) is selected for the creation of an RDF triple (involving the identified entities and property). Hence, this step implements a hybrid similarity measure composed of two types of measures, a corpus-based and a knowledge-based [23]. Hybrid measures demonstrated their effectiveness with regards to individual measures [24]. Details of the measures are presented in following subsections.

3.4.2.1 Knowledge-based measure. It calculates the semantic similarity between two words using information derived from semantic networks; being WordNet a popular such network [25, 26, 27]. The following aspects should be considered:

- Existing measures usually compare pairs of concepts (mental representations of a real-world object) to obtain a similarity score. Concepts in WordNet are organized as sets of synonyms called synsets. Thus, we obtained the synset of each word according to its POS and WordNet relevance.
- Most of the knowledge-based measures cannot be applied for words with distinct POS and thus, the same class must be used.

The general idea of the knowledge-based measure is shown in Eq. 1, where given two input sentences (s_1, s_2), the similarity score is returned.

$$sim(s_1, s_2) = \frac{\sum_{w \in s_1} maxSim(w, s_2)}{\sum_{w \in s_1} f(w, s_2)} \tag{1}$$

where

$$f(w, s_2) = \begin{cases} 1, & \text{if } maxSim(w, s_2) > 0 \\ 0, & \text{otherwise} \end{cases}$$

Consider that Eq. 1 calculates the semantic similarity between two given sentences by iterating over the words in each sentence. The process looks for the maximum similarity score of a word w in the first sentence (s_1) against every word in the second sentence (s_2). Finally, a weighted measure is obtained by considering only those cases that were able to indicate a similarity score (as supported by f). Note that this function is not symmetric and thus, it can be adapted in the following way $\frac{1}{2}(sim(s_1, s_2) + sim(s_2, s_1))$.

3.4.2.2 Corpus-based measure. This type of measures determine the degree of similarity between words using information modeled from large corpora. This step implements the Word Mover's Distance (WMD) [28], which is based on *word embeddings* (representations of words as dense vectors –embeddings– that are derived by diverse training methods based on neural networks following a Deep Learning approach) to calculate the distance between the embedded words of two documents (paragraphs or set of words). This measure works well with small documents (in our case comparing sentences), outperforms diverse measures in the literature (applied in clustering tasks) [28], and it is easily implementable through existing libraries [29]. The routine for applying the WMD distance starts by loading the model and then preparing the input sentences; deleting stop-words and splitting the text into tokens. After the similarity distance is obtained, it is determined if the distance is infinite (as provided in languages such as Python), which implies that the input sentences are totally different. Implementation details are later presented.

3.4.2.3 Overall function. This research work implements a linear combination of knowledge- and corpus-based measures as provided by Achananuparp *et al.* [24], which is described in Eq. 2. In this case, the value of α allows the function to control the weight of the knowledge-based measure, where a value of $\alpha = 0.5$ indicates equal importance for both measures. Note that kb denotes the knowledge-based measure and cb the corpus-based measure (implemented through WMD).

$$sim_{kb+cb}(s_1, s_2) = \alpha sim_{kb}(s_1, s_2) + (1 - \alpha) sim_{cb}(s_1, s_2) \tag{2}$$

It is worth mentioning that the score of the knowledge-based measure has a value between `0` and `1`, where a score of `1` represents identical text fragments. However, the corpus-based measure has a value between `0` and ∞, where `0` indicates identical text segments. Thus, we performed a normalization to combine these measures.

The disambiguation strategy is shown in Algorithm 1, where the algorithm receives as input the predicate (PRED) of a semantic relation and the set of property candidates (PROPCANDIDATES). Every property contains its URI, label, and parameters to put the similarity scores with respect to the predicate (line 2). On the other hand, the normalization of the corpus-based distance (line 15) considers the maximum distance of the set of property candidates against the predicates (outlier values omitted) to keep values between `0` and `1`, where `1` indicates identical text segments.

Algorithm 1 Disambiguation process.

Data: PRED, PROPCANDIDATES
Result: $Most\_similar\_property$

$maxDP \leftarrow 0$; /* Max. distance for normalization */
$similarProperty \leftarrow Map(url = null, label = null, alt = null, wmdistance = 0, wndistance = 0, weightDistance = 0)$; /* Property to be returned */
$\alpha \leftarrow 0.6$; /* parameter for combining measures */
forall $property \in$ PROPCANDIDATES **do** /* Iterate over properties */
  $property.wndistance \leftarrow sim_{kb}(\text{PRED}, property.label)$; /* knowledge-based measure */
  $property.wmdistance \leftarrow sim_{cb}(\text{PRED}, property.label)$; /* corpus-based measure */
  if $property.wmdistance > maxDP$ **then**
    $maxDP \leftarrow property.wmdistance$;
  end
end
forall $property \in$ PROPCANDIDATES **do** /* combined measure */
  if $property.wmdistance = -1$ **then** /* Dissimilar value set to 0 */
    $property.wmdistance \leftarrow 0$;
  else
    $property.wmdistance \leftarrow 1 - (property.wmdistance/maxDP)$; /* Normalization */
  end
  $property.weightDistance = (\alpha)property(.wndistance) + (1.0 - \alpha)property.wmdistance$;
  if $property.weightDistance > similarProperty.weightDistance$ **then** /* Check for most similar property */
    $similarProperty \leftarrow property$;
  end
end
return $similarProperty$;

3.5 Representation

This stage refers to the representation of the input entities and the property selected through the previous steps. Thus, RDF triples are represented within named graphs through the serialization format TriG. According to the example presented in Figure 1, a representation using the TriG format is presented in Listing 1, where only one triple is represented for demonstration purposes. Note that we still use fictitious URIs for readability but instead of using the entity wd:fcBarcelona, we included an NP-entity containing the entities wd:Barcelona, wd:soccer, wd:Team through the property `dcterms:isPartOf`.

Listing 1: RDF representation example using TriG

```
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix cvst:  <http://www.local.mx/> .
@prefix wd:  <http://www.wikidata.org/entity/> .
@prefix wdt:  <http://www.wikidata.org/prop/direct/> .

cvst:testing.html {
wd:Leo_Messi wdt:playsFor cvst:Barcelona_soccer_team .

wd:Barcelona dcterms:isPartOf cvst:Barcelona_soccer_team .
wd:Team dcterms:isPartOf cvst:Barcelona_soccer_team .
wd:soccer dcterms:isPartOf cvst:Barcelona_soccer_team .
}
```

4 Implementation details

This section provides details of an implementation of the proposed strategy. The Disambiguator step is entirely implemented in Python and the other stages in Java. Particular details of every step are presented as follows:

Preprocessing. Preprocessing tasks were performed through the Stanford CoreNLP tool [30], where models for English were used in the configuration.

EEL & grouping. Instead of using three EEL systems as mentioned in [22], we implemented an ensemble EEL system with four systems: DBpedia SpotLight [2], TagME [3], Babelfy [4], and WAT [5].

Relation Extraction. We extracted semantic relations from text through the ClausIE tool [31, 22].

Property Selection. Some particular aspects of this stage are:

- Entity selection. In case that more than one entity is contained within every argument (subject or object), two options can be applied: based on thematic roles (through SRL) or entity closeness. First, SRL allows the strategy to select the entities involved in the main roles of the relation (the causer and the undergoer of an action). However, in case that SRL does not provide information, entities close to the verb are selected (with a maximum distance of 2 words). Otherwise, the sentence is discarded.
- SPARQL queries were submitted to the Wikidata endpoint* through a Java application implementing the Jena library. Note that for time restrictions, a limit of 120,000 individuals was set to the query, which indicates that only the existing properties in those individuals are processed. This parameter was empirically selected to fit the timeout restriction of 60 seconds imposed by the Wikidata public endpoint.
- We retrieve properties from the Wikidata KB with features such as the URL, label (literal description), and alternative labels (additional descriptions).

*https://query.wikidata.org

Disambiguator. The hybrid similarity measure was implemented in Python. It is composed as follows:

- Knowledge-based measure. This measure was implemented through the NLTK library†.
- Corpus-based measure. WMD measure was implemented through the Gensim library [29].
- Invocation. Finally, an HTTP server was configured to allow the disambiguation module to be invoked via POST requests from the main Java application.

Note that the similarity measures and the configuration of the overall function were established according to the experiments performed in the following section.

Representation. This step was implemented in Java through the Jena library for generating TriG files.

It is worth mentioning that our focus is on presenting an implementation for the proposed method using traditional/available NLP tools. In this sense, latest and precise tools (for example those based on Deep Learning [32]) can be used instead of the mentioned for this implementation. However, training and configuring such an implementation is out of the scope of this work.

5 Experiments

The evaluation considers two aspects: the disambiguation strategy and the RDF representation. The former is evaluated through diverse similarity measures and models with the purpose of selecting the parameters used in the disambiguation of the final representation process. The latter was evaluated under two conditions; quantitative and qualitative. While the first condition allows us to collect the number of represented objects, the second condition is performed under an *a posteriori* assessment to check the quality of the represented information through the intervention of human judges. The following subsections present descriptions of the used datasets, metrics, and experiments.

5.1 Datasets and metrics

Three datasets were used for the experiments: LonelyPlanet [33] and BBC‡ for the RDF representation and Semantic Textual Similarity (STS) [34] for the disambiguation task. The dataset are described in Table 1.

Regarding the metrics, while the disambiguation strategy employs the Pearson correlation [35] to measure the semantic similarity between sentences, the RDF representation considers the Precision measure (proportion of retrieved elements that are correct) for a qualitative analysis [19] and the Fleiss Kappa measure [36] for assessing the inter-rater agreement among judges.

5.2 Parameter configuration for the disambiguation strategy

This subsection evaluates the disambiguation strategy to configure the parameters of the overall function used in the property selection step.

† https://www.nltk.org/
‡ http://mlg.ucd.ie/datasets/bbc.html

Table 1. Description of datasets

Dataset	Domain	Documents	Sentences
LonelyPlanet	Tourism	1801	16540
	Business	510	5988
	Entertainment	386	4482
BBC	Politics	417	5902
	Sport	511	6514
	Tech	401	6901
STS-Train	News, Forum	-	5749
STS-Test	News, Forum	-	1379

Scenario. The measures were evaluated as follows:

- Knowledge-based. Four measures were evaluated: LCH [27], WUP [25], PATH [37], and LIN [26]. Such measures were selected according to studies that demonstrate their performance [23].
- Corpus-based. The WMD measure was evaluated using five distinct models trained with documents of news and Wikipedia, which were proposed by Zuccon *et al.* [38]. The models are described in Table 2.
- Combined measure. This considers the evaluation of the best-scored measures in the two previous cases, combined under the overall function. In this case, the alpha value (α) was changed from `0.1` to `0.9` with intervals of `0.1`.

Table 2. Word embedding models used for the evaluation of the WMD measure. *Emb. Space* refers to the embedding space features used in the model.

Model	Type	Emb. Space	Dataset
WMD1	cbow	700	News, TREC-AP8889
WMD2	skip-gram	700	News, TREC-AP8889
WMD3	cbow	700	Wikipedia, 3billion words
WMD4	skip-gram	700	Wikipedia, 3billion words
Gnews	skip-gram	300	Google news

Results. The results are presented in Table 3, where the first four rows refer to knowledge-based measures and the last five to the WMD using diverse models (corpus-based). Note that for every measure (row), the Pearson correlation (Corr.) is depicted together with the execution time in seconds.

According to Table 3, the PATH and WMD4 measures got the highest results (Pearson correlation close to `1`) for each category. In this sense, both measures were selected to evaluate the combined measure on the two STS datasets (i.e., Test and Train). The evaluation results of the combined measure are shown in Figure 4, where *Corr.* represents the Pearson correlation and α refers to the alpha value used for the weighted overall function.

Table 3. Evaluation results of individual semantic similarity measures. **Corr.** refers to the Person correlation obtained for the whole dataset, and **Time** refers to the execution time in seconds.

Dataset	STS-test		STS-train	
Measure	**Corr.**	**Time**	**Corr.**	**Time**
LCH	0.4419	30.1322	0.4751	35.5923
WUP	0.4961	51.9969	0.4877	95.6836
PATH	**0.5505**	49.8073	**0.5594**	89.2397
LIN	0.4013	6426.0705	0.4102	26788.7501
WMD1	0.4284	62.7298	0.5025	85.0647
WMD2	0.4348	67.3631	0.5102	75.5461
WMD3	0.4428	151.6637	0.5073	157.5302
WMD4	**0.4565**	143.3102	**0.5277**	169.9340
Gnews	0.4380	84.3385	0.5202	116.6253

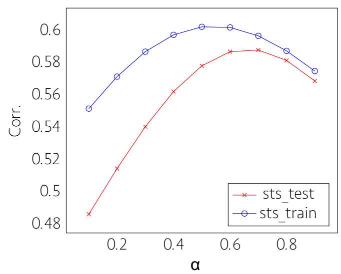

Figure 4. Evaluation results of the hybrid similarity measure. **Corr.** refers to the Person correlation, and α refers to the weight given to the measure.

Discussion. Although LIN has demonstrated that outperforms state of the art measures at word vs word similarity comparison [39], the results of this study demonstrate that the PATH measure (based on short distances) provided better performance at sentence vs sentence similarity. In general, we think this fact is given due to the used datasets and configured parameters, where LIN relies on Information Content (IC) to measure the specificity of a concept.

In summary, the disambiguation strategy relies on a hybrid measure composed of the PATH and WMD measures, which are based on WordNet information and word embeddings (respectively) that are combined through an overall function tuned with an α value of `0.6` (as the median of the best results). It is worth mentioning that the purpose of this step is not to propose a new disambiguation strategy but to configure the parameters for the used measures. In this sense, the results demonstrate that the hybrid measure outperforms all the individual measures tested in this evaluation.

5.3 Selection of property labels for disambiguation

Property candidates obtained from Wikidata contain diverse features that can be used as input data to the disambiguation strategy. Thus, we identified four possible ways the predicate can be compared against the property candidates:

- Option 1 (Simple case). The predicate is compared against each label of the property candidates (predicate vs labels).
- Option 2 (Enriched). This case considers an enrichment of the predicate and the property label with the Subject (S) and Object (O) of the semantic relation. Thus, the disambiguation strategy takes as input the enriched predicate (`S+predicate+O`) and each enriched property label (`S+label+O`).
- Option 3 (Iterative case). Some property candidates often contain alternative labels that indicate other possible ways to identify the property. In this case, we consider the comparison of the predicate phrase against each alternative label of every property candidate. It is similar to Option 1 but iterating also over alternative labels.
- Option 4 (Enriched iterative). This case is similar to Option 2 and 3, where the predicate is enriched with information of the semantic relation (`S+predicate+O`) and compared against an enriched version of each alternative label of each property candidate (`S+altLabel+O`).

Scenario. We considered the following criteria:

- Data sampling. A sample of 235 documents from the LonelyPlanet dataset was obtained through the strategy proposed by Krejcie and Morgan [40], where the input parameters considered a confidence of `90` and an error rate of 5.
- Data presentation. The represented triples were showed to the judges through a web application to be evaluated with three categories: *Good*, if the triple conveys correctly the sentences; *Fair*, if the triple conveys meaningful information; and *Poor*, if the triple is incorrect. Thus, every element of the triple was reviewed, but only if all elements are correct (Good) the triple is deemed as correct. This is because the experiment is focused on the evaluation of the final output considering the diverse input configurations. Thus, every element is evaluated according to the categories as follows:
  - Poor: If the element to be judged is not syntactically acceptable/correct or does not make any sense (it does not convey something useful or it is incoherent).
  - Fair: If the element does not refer specifically to the idea presented in the original sentence or semantic relation but it is acceptable (makes sense, convey a valid statement or coherent idea).
  - Good: If the element is correct and exactly or semantically convey the idea of the original sentence/semantic relation.

  These criteria are also used in the qualitative evaluation (Section 5.5).
- Profile of judges. Two human judges with notions of the domain, model, and language participated in the evaluation. The evaluation was performed at the same time, where both judges discussed and assessed the same output together.

Results. A total of 239 triples were extracted and represented from the 235 input documents. The results of the evaluation are presented in Figure 5, where **Dis. Options** refers to the disambiguation options, and **Cases** refers to the number of assessed triples for each category. Note that the same number of triples is evaluated for each option, where the component that particularly changes from triples between options is the property selected by the disambiguation input. Thus, the judges evaluated a total of 956 RDF triples (accumulated from the four options).

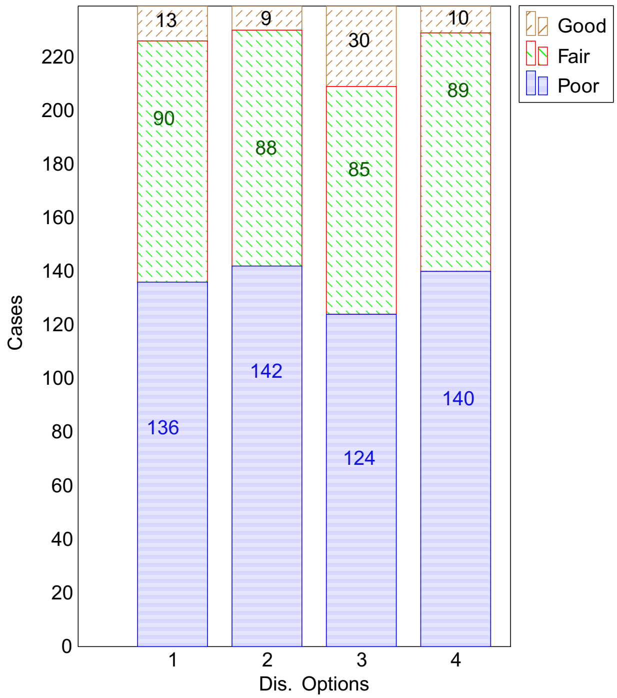

Figure 5. Results of the RDF representation evaluation using four distinct disambiguation inputs.

According to the results shown in Figure 5, the precision was calculated in two distinct ways, in which Strong and Weak configurations take only "Good" records or "Good+Fair" records respectively. The results of the precision are shown in Table 4, where *Prec.* refers to the type of precision (Weak or Strong) and *Opt.* refers to the disambiguation option.

Table 4. Precision results of the RDF representation using four distinct disambiguation inputs.

Prec./Opt.	**1**	**2**	**3**	**4**
Weak	0.4310	0.4059	**0.4812**	0.4142
Strong	0.0544	0.0377	**0.1255**	0.0418

Discussion. The option 3 (Iterative case) provided the best result in the evaluation due to the availability of diverse labels that describe the same property. However, due to the short context of the compared elements, the NLP tools not always provide correct tags, which has repercussions for identifying the concept in the knowledge-based measures. Moreover, even if diverse property candidates are collected, it is possible that none of them are associated with the predicate of the input semantic relation. Thus, the configuration of parameters takes into account only the precision of triples, where a different configuration strategy (e.g., trade-off between precision and recall) might require more benchmark datasets (scarcely available) with standardized data, which are hard to produce at this moment.

This experiment provided the final configuration of the disambiguation strategy, which was used for the representation of RDF triples.

5.4 Quantitative evaluation

The aim of this experiment is to analyze the number of elements extracted and represented from sentences into RDF triples.

Scenario. LonelyPlanet and BBC datasets were used. The parameters of the property selector were configured as presented in Section 4.

Results. The results are presented in Table 5, where **Rep. Sent.** refers to the number of represented sentences, **Triples** refers to the total number of triples counting associations of entities and label descriptions, **Binary statements** refers to the main triples represented (those with a property selected from Wikidata), **Entities** and **NP-entities** refers to the represented named entities.

Table 5. Quantitative results of the representation.

Dataset	Rep. Sent.	Triples	Binary statements	Entities	NP-entities
LonelyPlanet	2013	14226	2307	6543	2780
BBC (business)	1137	9468	1652	3941	1937
BBC (entertainment)	712	6256	1038	2705	1223
BBC (politics)	1138	8842	1605	3957	1703
BBC (sport)	851	6718	1180	3019	1180
BBC (tech)	1136	10360	1897	4453	2065
Total	**6987**	**55870**	**9679**	**24618**	**10888**

Discussion. The strategy was able to represent only a fraction of the initial input sentences. The best case was for the BBC dataset regarding the business topic with 27.58% of input sentences represented and the worst case for the LonelyPlanet dataset with only 13.94%. This fact demonstrates diverse problems produced in early stages of the representation strategy (preprocessing and knowledge extraction tasks). Additionally, the percentage of represented triples depends on diverse factors such as the lack of relations (from the sentence), properties (from the KB), and aspects related to the grammatical tense of sentences that cause a misinterpretation of the triple elements, where the best results are produced when the subject directly involves an action over the object.

5.5 Qualitative evaluation of binary statements

This experiment evaluates the quality of the represented RDF triples. Due to the lack of gold-standard datasets, the experiment is performed by an *a posteriori* revision, where human judges evaluate a set of represented triples.

Scenario. The experiment was based on the one presented in Section 5.3, where the judges evaluated a sample of RDF triples. However, the presentation of data to the human judges was conducted using the *Figure-Eight* (`www.figure-eight.com`) crowdsourcing platform, which allows the collaboration of users around the world with a controlled quality at a determined price/monetary cost. Details of the experiment are presented as follows:

- Data sampling. A sample of 250 RDF triples represented from the BBC dataset was obtained using a confidence level of 90 and an error margin of 5.
- Presentation of data. The RDF triples and the semantic relation were presented to the judges through a web application provided by the *Figure-Eight* platform, which consisted of the following elements:
  - Description. A background of the RDF model, instructions, and examples for judging records.
  - Records. The RDF triples are presented to the judges by means of web forms, each one containing the elements to evaluate (i.e., subject, predicate, object, and semantic relation).
  - Evaluation categories. This experiment considers three categories for assessing the records by the judges: *Poor*, *Fair*, and *Good*.
  - Quality. An initial set of 40 RDF "gold" records were used as a reference by the platform. In case that one gold record is wrongly evaluated by the judge, the quality of the judge is penalized and the correct answer is exposed as feedback. Only those judges that answer correctly to the 70% of the gold records were accepted for the evaluation.

An example web form (including comments) is presented in Figure 6. In this case the fragment to be evaluated is only the semantic relation. The original sentence is included as a supporting element, which helps the judge to evaluate the element. Note that, although only the semantic relation is evaluated in this part, the image of the RDF graph is included to facilitate the evaluation of the complete record.

In addition to the example form presented in Figure 6, we provide two evaluation examples for each of the three specified categories (Poor, Fair, Good) in Figure 7, where the first element is a sentence, then a semantic relation and finally the triple in form of RDF graph. We use a graph notation because the subject or object elements might be composed of more than one entity. In general, Poor quality triples are depicted due to problems with the extraction of entities and relations derived from lengthy sentences (diverse elements involved on it).

Results. The precision results are presented in Table 6. Likewise, given the three categories for assessing the records, the precision is obtained in two configurations, Strong and Weak. It is worth mentioning that six judges were required for evaluating three times each of the records. This is because the quality and design usability limit the number of records assigned to each judge. On the other hand, the inter-rater agreement values of this evaluation are shown in Table 7, in which the parameters consisted of three raters (every record was evaluated three times), 250 cases (RDF triples), and three categories (Poor, Fair, and Good).

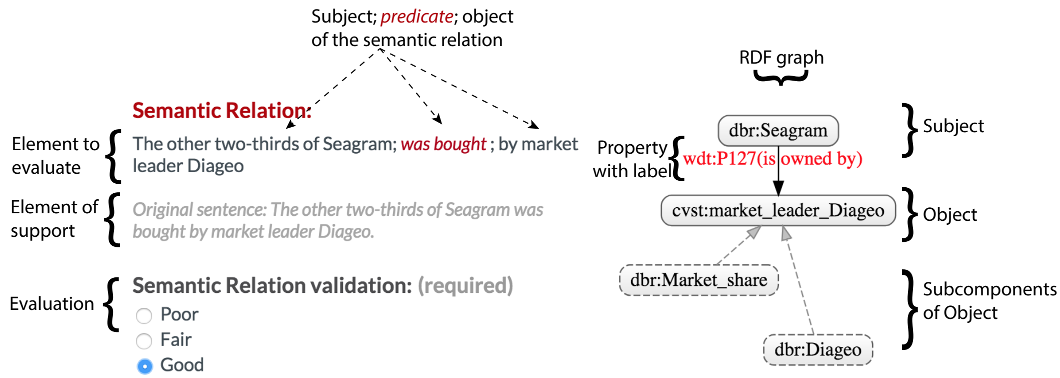

Figure 6. Evaluation form designed through the *Figure-Eight* platform.

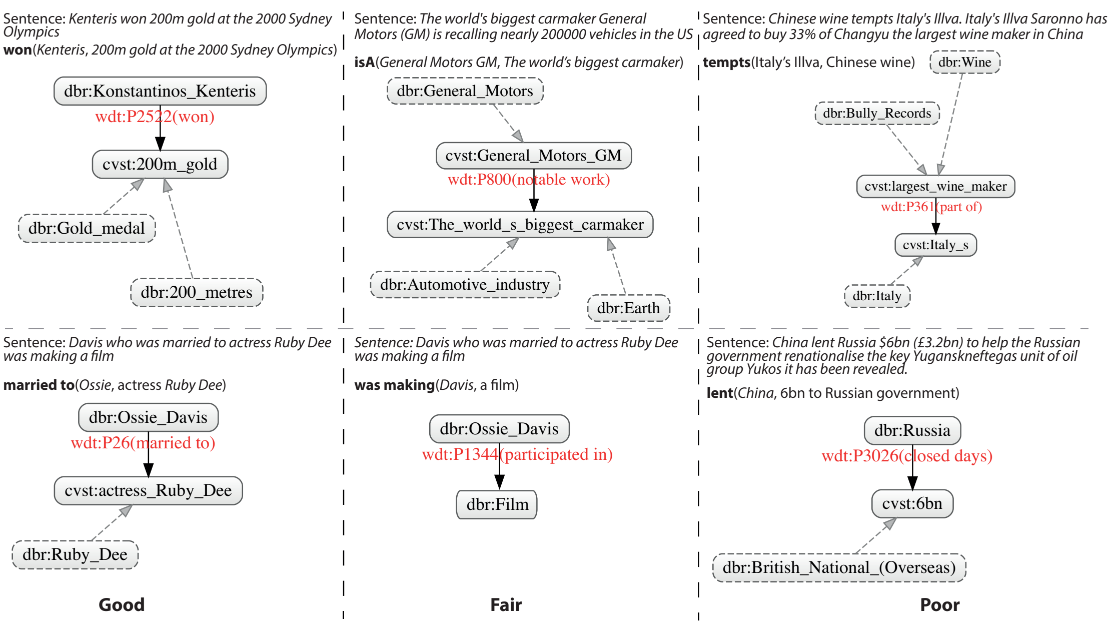

Figure 7. Examples of evaluation for each of the three defined categories (Poor, Fair, and Good).

Table 6. Precision of extracted elements by the RDF representation.

Prec./Elem.	Subject	Property	Object	Relation	Triple
Strong	0.712	0.132	0.527	0.485	0.107
Weak	**0.892**	**0.489**	**0.823**	**0.871**	**0.596**

Table 7. Inter-rater agreement for the evaluation of RDF triples.

Elem./Measure	Overall	Kappa
Subject	0.685	0.528
Property	0.631	0.446
Object	0.571	0.356
Relation	0.521	0.282

Discussion. While the Strong configuration demonstrates how accurate the statements represent the original input sentences, the Weak configuration allows a more permissive evaluation that accepts novel facts derived from such sentences. Hence, the following aspects were observed:

- Entity linking. The task for the extraction and linking of entities obtained better results than other tasks (seen for the association of *subject* and *object*). This fact demonstrates the maturity of the task, which has been thoroughly studied by the community.
- Property selection. The precision obtained by both configurations demonstrate very contrasting results, which are produced for several factors (and language variations) such as the recognition of elements (by NLP tools), availability of property candidates (in the KB), matching by similarity measures, verbal tenses (and negation), and so on.
- Inter-rater agreement. It can be noted that the inter-rater agreement values presented in Table 7 demonstrate variations. This is due to the subjective answers provided by the judges. However, the results show an agreement from fair to moderate [41], which demonstrate that the evaluation was not guided by chance.

5.6 Qualitative comparison with an existing approach

This experiment is aimed at comparing the proposed method with an existing approach for the representation of RDF triples.

Scenario. We implemented the approach proposed by Exner and Nugues [14], which is based on predicate–property patterns to associate semantic relations to entities and properties from a KB. Thus, the experiment was configured with the following data:

- Entities were recognized through DBpedia Spotlight.
- Semantic relations were recognized through MateTools (SRL), where arguments were matched against the found entities.
- For property matching, a total of `84,063` patterns were obtained from Wikipedia articles as presented by [42]. Each pattern consists of a textual predicate and an associated RDF triple (subject-predicate-object). Thus, found entities and the relation are matched against the patterns.
- The evaluation was performed using the Gold Standard presented by Kertkeidkachorn and Ichise [42], consisting of 100 sentences randomly selected from Wikipedia. Each sentence presents its corresponding triple(s) linked to DBpedia resources. Note that we only consider triples with object properties.

Results. The obtained Precision, Recall, and F-1 measures are presented in Table 8.

Table 8. Result of comparison with an existing approach

Approach	Precision	Recall	F-1
Exner and Nugues [14]	0.50	0.10	0.16
Proposed method	**0.65**	**0.17**	**0.27**

Discussion. The results demonstrated that our method overcomes the output provided by the pattern-based method of Exner and Nugues [14]. This fact is presented because our method is not guided by a set of patterns, which might not contain the property associated with the entities at hand. Although our method also overcomes the precision presented by Kertkeidkachorn [42] (`0.54`), we did not include it in Table 8 because we only take into account object properties in the comparison.

6 Conclusions

This paper presented a method for the information representation through RDF triples extracted from sentences. The idea is to extract entities and their semantic relation from text to later associate them to resources and properties from a KB. The experiments demonstrate that the representation of RDF statements is still a challenging task. The results obtained by the experiments give an idea of the application of the proposed method, which would be useful in tasks such as semantic enrichment or to guide users in the semantic annotation of sentences. It is worth mentioning that the components configured in every step of the method are replaceable and thus, the proposed solution served to prove the capability to represent information without depending on user intervention or pre-trained modules. Of course, establishing restrictions to the sort of representations and using ad-hoc tools (e.g., recent proposals based on Deep Learning) would improve the precision of the whole system because, as mentioned, many of the issues of the final representation are often inherited from early stages of the process.

As future work, we plan to develop a versatile method implementation following our proposed representation strategy, incorporating different NLP tools/solutions. For example, those based on deep learning, where large datasets and model representations are often required for training and parameter configuration.

Acknowledgements

This work was funded in part by SEP-Cinvestav, Mexico project number 229.

References

[1] Vrandecic D and Krötzsch M. Wikidata: a free collaborative knowledgebase. *Communications of the ACM* 2014; 57(10): 78–85. DOI:10.1145/2629489.

[2] Mendes PN, Jakob M, García-Silva A et al. DBpedia spotlight: shedding light on the web of documents. In *International Conference on Semantic Systems, 2011*. ACM, pp. 1–8. DOI: 10.1145/2063518.2063519.

[3] Ferragina P and Scaiella U. Tagme: On-the-fly annotation of short text fragments (by Wikipedia entities). In *Conference on Information and Knowledge Management, 2010*. ACM, pp. 1625–1628.

[4] Moro A, Raganato A and Navigli R. Entity Linking meets Word Sense Disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics* 2014; 2: 231–244.

[5] Piccinno F and Ferragina P. From TagME to WAT: A New Entity Annotator. In Carmel D, Chang M, Gabrilovich E et al. (eds.) *International Workshop on Entity Recognition & Disambiguation, 2014*. ACM, pp. 55–62.

[6] Mintz M, Bills S, Snow R et al. Distant supervision for relation extraction without labeled data. In Su K, Su J and Wiebe J (eds.) *Annual meeting of the Association for Computational Linguistics, 2009*. ACL, pp. 1003–1011.

[7] Dutta A, Meilicke C and Stuckenschmidt H. Enriching structured knowledge with open information. In Gangemi A, Leonardi S and Panconesi A (eds.) *World Wide Web Conference, 2015*. ACM, pp. 267–277.

[8] Martinez-Rodriguez JL, Hogan A and Lopez-Arevalo I. Information extraction meets the semantic web: A survey. *Semantic Web Journal (to appear)* 2018; .

[9] Fossati M, Dorigatti E and Giuliano C. N-ary relation extraction for simultaneous T-Box and A-Box knowledge base augmentation. *Semantic Web* 2018; 9(4): 413–439. DOI:10.3233/SW-170269.

[10] Nguyen TVT and Moschitti A. End-to-end relation extraction using distant supervision from external semantic repositories. In *Annual meeting of the Association for Computational Linguistics: Human Language Technologies, 2011*. ACL, pp. 277–282.

[11] Augenstein I, Maynard D and Ciravegna F. Distantly supervised Web relation extraction for knowledge base population. *Semantic Web* 2016; 7(4): 335–349. DOI:10.3233/SW-150180.

[12] Wang Z, Chang B and Sui Z. Distantly supervised neural network model for relation extraction. In *Chinese Computational Linguistics, 2015*. Springer, pp. 253–266.

[13] Geng Z, Chen G, Han Y et al. Semantic relation extraction using sequential and tree-structured LSTM with attention. *Information Sciences* 2020; 509: 183–192.

[14] Exner P and Nugues P. Entity extraction: From unstructured text to DBpedia RDF triples. In *The Web of Linked Entities Workshop, 2012*. CEUR-WS, pp. 58–69.

[15] Niu F, Zhang C, Ré C et al. DeepDive: Web-scale Knowledge-base Construction using Statistical Learning and Inference. In Brambilla M, Ceri S, Furche T et al. (eds.) *International Workshop on Searching and Integrating New Web Data Sources, 2012*. CEUR-WS.org, pp. 25–28.

[16] Etzioni O, Banko M, Soderland S et al. Open information extraction from the web. *Commun ACM* 2008; 51(12): 68–74. DOI:10.1145/1409360.1409378.

[17] Liu F, He S, Liu S et al. Open relation mapping based on instances and semantics expansion. In Banchs RE, Silvestri F, Liu T et al. (eds.) *Asia Information Retrieval Societies Conference, 2013*. Springer, pp. 320–331.

[18] Soderland S and Mandhani B. Moving from textual relations to ontologized relations. In *AAAI, 2007*. pp. 85–90.

[19] Dutta A, Meilicke C and Stuckenschmidt H. Semantifying triples from open information extraction systems. In Endriss U and Leite J (eds.) *European Starting AI Researcher Symposium, 2014*. IOS Press, pp. 111–120.

[20] Presutti V, Nuzzolese AG, Consoli S et al. From hyperlinks to Semantic Web properties using Open Knowledge Extraction. *Semantic Web* 2016; 7(4): 351–378. DOI:10.3233/SW-160221.

[21] Gerber D, Hellmann S, Bühmann L et al. Real-Time RDF Extraction from Unstructured Data Streams. In *International Semantic Web Conference (ISWC)*, volume 8218. Springer, 2013. pp. 135–150. DOI:10.1007/978-3-642-41335-3_9.

[22] Martinez-Rodriguez JL, Lopez-Arevalo I and Rios-Alvarado AB. OpenIE-based approach for Knowledge Graph construction from text. *Expert Systems with Applications* 2018; 113: 339–355. DOI:10.1016/j.eswa.2018.07.017.

[23] Mihalcea R, Corley C and Strapparava C. Corpus-based and knowledge-based measures of text semantic similarity. In *National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, 2006*. AAAI Press, pp. 775–780.

[24] Achananuparp P, Hu X and Shen X. The evaluation of sentence similarity measures. In Song I, Eder J and Nguyen TM (eds.) *Data Warehousing and Knowledge Discovery, 2008*. Springer, pp. 305–316.

[25] Wu Z and Palmer MS. Verb semantics and lexical selection. In Pustejovsky J (ed.) *Annual Meeting of the Association for Computational Linguistics, 1994*. Morgan Kaufmann Publishers, pp. 133–138.

[26] Lin D. An information-theoretic definition of similarity. In Shavlik JW (ed.) *International Conference on Machine Learning, 1998*. Morgan Kaufmann, pp. 296–304.

[27] Leacock C and Chodorow M. Combining local context and wordnet similarity for word sense identification. *WordNet: An electronic lexical database* 1998; 49(2): 265–283.

[28] Kusner MJ, Sun Y, Kolkin NI et al. From word embeddings to document distances. In Bach FR and Blei DM (eds.) *International Conference on Machine Learning, 2015*. pp. 957–966.

[29] Rehurec R and Sojka P. Software framework for topic modelling with large corpora. In *New Challenges for NLP Frameworks, 2010*. University of Malta, pp. 46–50. DOI:10.13140/2.1.2393.1847.

[30] Manning CD, Surdeanu M, Bauer J et al. The Stanford CoreNLP Natural Language Processing Toolkit. In *Annual meeting of the Association for Computational Linguistics, 2014*. pp. 55–60.

[31] Del Corro L and Gemulla R. Clausie: Clause-based open information extraction. In *International Conference on World Wide Web, 2013*. ACM, pp. 355–366.

[32] Young T, Hazarika D, Poria S et al. Recent trends in deep learning based natural language processing [review article]. *IEEE Computational Intelligence Magazine* 2018; 13(3): 55–75.

[33] Cimiano P. *Ontology learning and population from text - algorithms, evaluation and applications*. Springer, 2006. ISBN 978-0-387-30632-2.

[34] Cer DM, Diab MT, Agirre E et al. Semeval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *11th International Workshop on Semantic Evaluation, Vancouver, Canada, 2017*. ACL, pp. 1–14.

[35] Goshtasby AA. Similarity and dissimilarity measures. In *Image registration*. Springer, 2012. pp. 7–66.

[36] Randolph JJ. Free-marginal multirater kappa (multirater k [free]): An alternative to fleiss' fixed-marginal multirater kappa. *Joensuu Learning and Instruction Symposium* 2005; .

[37] Pedersen T, Patwardhan S and Michelizzi J. Wordnet: : Similarity - measuring the relatedness of concepts. In *Proceedings of the Nineteenth National Conference on Artificial Intelligence, Sixteenth Conference on Innovative Applications of Artificial Intelligence, 2004*. AAAI Press / The MIT Press, pp. 1024–1025.

[38] Zuccon G, Koopman B, Bruza P et al. Integrating and evaluating neural word embeddings in information retrieval. In Park LAF and Karimi S (eds.) *Australasian Document Computing Symposium, 2015*. ACM, pp. 12:1–12:8.

[39] Budanitsky A and Hirst G. Evaluating WordNet-based measures of lexical semantic relatedness. *Computational Linguistics, 2006* 2006; 32(1): 13–47. DOI:10.1162/coli.2006.32.1.13.

[40] Krejcie RV and Morgan DW. Determining sample size for research activities. *Educational and psychological measurement* 1970; 30(3): 607–610.

[41] Landis JR and Koch GG. The measurement of observer agreement for categorical data. *biometrics* 1977; : 159–174.

[42] Kertkeidkachorn N and Ichise R. T2KG: an end-to-end system for creating knowledge graph from unstructured text. In *The Workshops of the The Thirty-First AAAI Conference on Artificial Intelligence, 2017*. AAAI Press, pp. 743–749.