

Evaluating Uncertainty in Textual Document

Fadhela Kerdjoudj^{1,2} and Olivier Curé^{1,3}

¹ University of Paris-Est Marne-la-vallée, LIGM, CNRS UMR 8049, France,
{fadhela.kerdjoudj, ocure}@univ-mlv.fr

² GeolSemantics, 12 rue Raspail 74250, Gentilly, France.

³ Sorbonne Universités, UPMC Univ Paris 06, LIP6, CNRS UMR 7606, France

Abstract. In this work, we consider that a close collaboration between the research fields of Natural Language Processing and Knowledge Representation becomes essential to fulfill the vision of the Semantic Web. This will permit to retrieve information from vast amount of textual documents present on the Web and to represent these extractions in an amenable manner for querying and reasoning purposes. In such a context, uncertain, incomplete and ambiguous information must be handled properly. In the following, we present a solution that enables to qualify and quantify the uncertainty of extracted information from linguistic treatment.

1 Introduction

Textual documents abound on the World Wide Web but efficiently retrieving information from them is hard due to their natural language expression and unstructured characteristics. Indeed, the ability to represent, characterize and manage uncertainty is considered as a key factor for the success of the Semantic Web [12]. The accurate and exhaustive extraction of information and knowledge is nevertheless needed in many application domains, *e.g.*, in medicine to comprehend the meaning of clinical reports or in finance to analyze the trends of markets. We consider that together with techniques from Natural Language Processing (NLP), best practices encountered in the Semantic Web have the potential to provide a solution to this problem. For instance, NLP can support the extraction of named entities as well as temporal and spatial aspects, while the Semantic Web is able to provide an agreed upon representation as well as some querying and reasoning facilities. Moreover, by consulting datasets from Linked Open Data (LOD), *e.g.*, DBpedia, Geonames, we can enrich the extracted knowledge and integrate it to the rest of the LOD.

The information contained in Web documents can present some imperfection, it can be incomplete, uncertain and ambiguous. Therefore, the texts content can be called into question, it becomes necessary to qualify and possibly quantify these imperfections to present to the end user a trusted extraction. However, qualification or quantification is a difficult task for any software application. In this paper, we focus on the uncertainty aspect and trustworthiness of the provided information in the text. A special attention of our work has been devoted

to representing such information within the Resource Description Framework (RDF) graph model. The main motivation being to benefit from querying facilities, *i.e.*, using SPARQL.

Usually, uncertainty is represented using reification, but this representation failed in representing uncertainty on triple property. Indeed, the reification does not identify which part of the triple (subject, predicate or the object) is uncertain. Here, we intend to manage these cases of uncertainties, as expressed in Example 1, while in the first sentence, the uncertainty concerns all the moving action (including, the agent, the destination and the date), in the second, the author expressed an uncertainty only on the date of the moving.

Example 1. 1. The US president *probably visited* Cuba this year.
2. The US president visited Cuba, *probably this year*.

We based our approach on an existing system developed at GEOLSemantics⁴, a french startup with expertise in NLP. This framework mainly consists of a deep morphosyntactic analysis and an RDF triple creation using trigger’s detection. Triggers are composed of one or several words (nouns, verbs, *etc.*) that represent a semantic unit denoting an entity to extract. For instance, the verb ”go” denotes a Displacement. The RDF graph obtained complies with an ontology built manually to support different domains such as Security and Economics. Actually, our framework consists of a set of existing vocabularies (such as Schema.org⁵, FOAF⁶, Prov⁷) to enrich our own main ontology, denoted `geo1`. This ontology contains the general classes which are common to many domains:

- *Document* : Text, Sentence, Source, DateIssue, PlaceIssue, *etc.*
- *Named entities* : Person, Organization, Location, *etc.*
- *Actions* : LegalProceeding, Displacing, *etc.*
- *Events* : SocialEvent, SportEvent, FamilialEvent, *etc.*

The contributions of this paper are two-fold: (1) We present a fine-grained approach to quantify and qualify the uncertainty in the text based on uncertainty markers; (2) We present an ontology which handles this uncertainty both at the resource and property level. This representation of uncertainty can be interrogated with a rewriting of SPARQL query.

The paper is organized as follows. Section 2 describes related work to uncertainty handling in Semantic Web. In Section 3, we present how to spot uncertain information in the text using specific markers. In Section 4, we propose an RDF-based representation of uncertainty in knowledge extraction. In Section 5, a use case is depicted with some SPARQL queries. Finally, we conclude in Section 6.

⁴ <http://www.geolsemantics.com/>

⁵ <http://schema.org/docs/schemaorg.owl>

⁶ <http://xmlns.com/foaf/spec/>

⁷ <http://www.w3.org/TR/prov-o/>

2 Related work

Integration of imprecise and uncertain concepts to ontologies has been studied for a long time by the Semantic Web community [13]. To tackle this problem, different frameworks have been introduced: Text2Onto [4] for learning ontologies and handling imprecise and uncertain data, BayesOWL [7] based on Bayesian Networks for ontologies mapping. In [6], the authors propose a probabilistic extension for OWL with a Bayesian Network layer for reasoning. Actually, fuzzy OWL [2, 20] was proposed to manage, in addition to uncertainty, some other text imperfection (such as imprecision and vagueness) with the help of fuzzy logics. Moreover, W3C Uncertainty Reasoning for the World Wide Web Incubator Group (URW3-XG) [12] describes an ontology to annotate uncertain and imprecise data. This ontology focuses on the representation of the nature, the model, the type and the derivation of uncertainty. This representation is really interesting but unfortunately does not show how to link the uncertainty to the concerned knowledge described in the text.

However, in all these works, the uncertainty was considered as a metadata. The ontologies which handle uncertainty are proposed to either create a fuzzy knowledge base (fuzzy ABox, fuzzy TBox, fuzzy Rbox) or to associate each class of the ontology to a super class which denotes the uncertain or fuzzy concept. To each axiom is associated a truth degree in $[0,1]$. Therefore, the user is required to handle two knowledge bases in parallel. The first one is dedicated to certain knowledge whereas the second is dedicated to uncertain knowledge. This representation could induce some inconsistencies between the knowledge bases. From a querying perspective this representation is also not appealing since it forces the user to query both bases and then combine the results. In order to avoid these drawbacks, we propose in this paper, a solution to integrate uncertain knowledge to the rest of the extraction. The idea is to ensure that all extracted knowledge, either be it certain or uncertain, is managed within the same knowledge base. This approach aims at ensuring the consistency of the extracted knowledge and eases its querying.

Moreover, it is worth noting that linguistic processing carried out on uncertainty management notably, Sauri[19] and Rubin [17][18] works, they payed attention to different modalities and polarity to characterize uncertainty/certainty.

The first one [19], considers two dimensions. Each event is associated to a factual value represented as a tuple $\langle mod, pol \rangle$ where *mod* denotes modality and distinguishes among: *certain*, *probable*, *possible* and *unknown*, *pol* denotes polarity values which are *positive*, *negative* and *unknown*.

In [17][18] four dimensions have been considered:

- *certainty level*: absolute, high, moderate or low.
- *author perspective*: if it is his/her point of view or a reported speech.
- *focus*: if it is an abstract information (opinion, belief, judgment...) or a factual one (event, state, fact...).
- *time*: past, present, future.

This model is more complete even if it does not handle negation. However, the authors do not explain how to combine all these dimensions to get a final interpretation to a given uncertainty. In this paper, we explain how to detect uncertainty in textual document and how to quantify it to get a global interpretation.

3 Uncertainty detection in the text

The Web contains a huge number of documents from heterogeneous sources like forums, blogs, tweets, newspaper or Wikipedia articles. However, these documents cannot be exploited directly by programs because they are mainly intended for humans. Before the emergence of the Semantic Web, only human beings could access the necessary background knowledge to interpret these documents. In order to get a full interpretation of the text content, it is necessary to consider the different aspects of the given information. Some piece of information can be considered as “perfect” only if it contains precise and certain data. This is rarely the case even for a human reader with some context knowledge. Indeed, the reliability of the data available on the Web often needs to be reconsidered, uncertainty, inconsistency, vagueness, ambiguity, imprecision, incompleteness and others are recurrent problems encountered in data mining. According to [9] the information can be classified into two categories : *subjective* and *objective*. An information is objective or quantitative if it indicates an observable, *i.e.*, something which is able to be counted for example. The other category is the subjective (qualitative) information. It can describe the opinion of the author, he may express his own belief, judgment, assumption, etc. Therefore, the second one is subject to contain imperfect data. Then, it becomes necessary to incorporate these imperfections within the representation of the extracted information.

In this paper, we are interested in the uncertainty aspect. In domains such as information theory, knowledge extraction and information retrieval, the term uncertainty refers to the concept of being unsure about something or someone. It denotes a lack of conviction. Uncertainty is a well studied form of data imperfection, but it is rarely considered at the knowledge level during extraction processing. Our approach consists in considering the life cycle of the knowledge from the data acquisition to the final RDF representation steps, *i.e.*, generating and persisting the knowledge as triples.

Evaluating uncertainties in text

As previously explained, the text may contain several imperfections which can affect the trustworthiness of an extracted action or event. So, during the linguistic processing, we need to pay attention to the modalities of the verb which indicate how the action or the event had happened, or how it will. Actually, the text provides information about the epistemic stance of the author, that he often commits according to his knowledge, singular observation or beliefs [16]. Moreover, natural languages offer several ways to express uncertainty, usually

expressed using linguistic qualifiers. According to [14, 8, 1] uncertainty qualifiers can be classified as follows:

- verbal phrases *e.g.*, *as likely as, chances are, close to certain, likely, few, high probability, it could be, it seems, quite possible.*
- expression of uncertainty with quantification *all, most, many, some, etc.*,
- modal verbs *e.g.*, *can, may, should.*
- adverbs, *e.g.*, *roughly, somewhat, mostly, essentially, especially, exceptionally, often, almost, practically, actually, really.*
- speculation verbs *e.g.*, *suggest, suppose, suspect, presume.*
- nouns *e.g.*, *speculation, doubt, proposals.*
- expressions *e.g.*, *raise the question of, to the best of our knowledge, as far as I know.*

All these markers help to detect and identify the uncertainty with different intensities. This helps in evaluating the confidence degree associated to the given information. For example : *it **may** happen* is less certain that *it will **probably** happen*. It is also necessary to consider modifiers such as *less, more, very*. Depending on the polarity of each modifier we add or subtract a predefined real number α , set to 0.15 in our experiment, to the given marker’s degree. We base our approach on a natural language processing. This processing indicates syntactic and semantic dependencies between words. From these dependencies we can identify the scope of each identifier in the text. Once these qualifiers are identified, the uncertainty of the knowledge can be specified and then quantified. By quantifying, we mean attributing a confidence degree which indicates how much we can trust the described entity. To this end, we associate to each marker a probabilistic degree. We defined three levels of certainty: (i) high=0.75, (ii) moderate=0.50, (iii) low=0.25. Moreover, we also base this uncertainty quantification on previous works in this field such as [3, 11] which define a mapping between the confidence degree and each uncertainty marker. This mapping is called Kent’s Chart and Table 1 provides an extract of it.

Table 1. Table of Kent’s Chart for expressions of degrees of uncertainty

Expression	Probability Degree
certain	100
almost certain, believe, evident, little doubt	85-99
fairly certain, likely, should be, appear to be	60-84
have chances	40-59
probably not, fairly uncertain, is not expected	15-39
not believe, doubtful, not evident	1-14

However, uncertainty markers are not the only way to generate uncertainty. Reported speech and future timeline are also considered as uncertainty sources.

These will be taken into account when the final uncertainty weight will be calculated. We notice that the trust of the reported speech depends of different parameters which affect the trust granted to its content:

- *the author of the declaration*: if the author name is cited, if the author has an official role (prosecutor, president...).
- *the nature of the declaration*: if it is an official declaration, a personal opinion, a rumor...

Example 2. A crook who burglarized homes and crashed a stolen car remains on the loose, but he **probably** left Old Saybrook by now, **police said** Thursday.

In Example 2, we can identify two forms of uncertainty. First, the author explicitly expresses, using the term (*probably*), an uncertainty about the fact that the crook left the city. The second one is related to the reported speech which comes from the police and is not assumed to be a known fact.

Therefore, for a given information described in the text, many sources of uncertainty can occur, then, it is necessary to combine all these uncertainties in order to get a final confidence degree to be attributed to the extracted information. With regard to this issue, we chose a Bayesian approach to combine all uncertainties to the concerned information. Indeed Bayesian network are well suited to our knowledge graph which is a directed acyclic graph. This choice is also motivated by the dependency that exists between children of uncertainty nodes. Indeed, to calculate the final degree of uncertain information, we need to consider its parents, if they contain uncertainty, then the conditional probability related to this parent is reverberated on the child.

4 RDF representation of uncertainty

In order to extract complete and relevant knowledge, we consider the uncertainty as an integral part of the knowledge instead of integrating it as an annotation. Usually, uncertainty is added as assertions to triples (the uncertainty assigned to each extracted knowledge). So, we represent it with some reification as recommended by [5]. Nevertheless, we encountered some difficulties to represent uncertainty on triples' predicates, as opposed to the whole triple. In the second sentence of Example 1, the uncertainty does not concern the whole moving but only its date. Only one part of the event is uncertain and the RDF representation has to take this into account. In fact, we cannot indicate using reification which part of the triple is uncertain, as shown in Figure 1, with reification, we give the same representation to both sentences in Example 1 even if they express different information. Indeed, reified statements cannot be used in semantic inferences, and are not asserted as part of the underlying knowledge base [21]. The reified statement and the triple itself are considered as different statements. So, due to its particular syntax (`rdf:Statement`) the reified triple can hardly be related to other triples in the knowledge base [15]. Moreover, using blank node to identify the uncertain statement prevents from obtaining good performance [10]. Indeed,

writing queries over RDF data sets involving reification becomes inconvenient. Especially, for one to refer to a reified triple, we need to use four additional triples linked by a blank node.

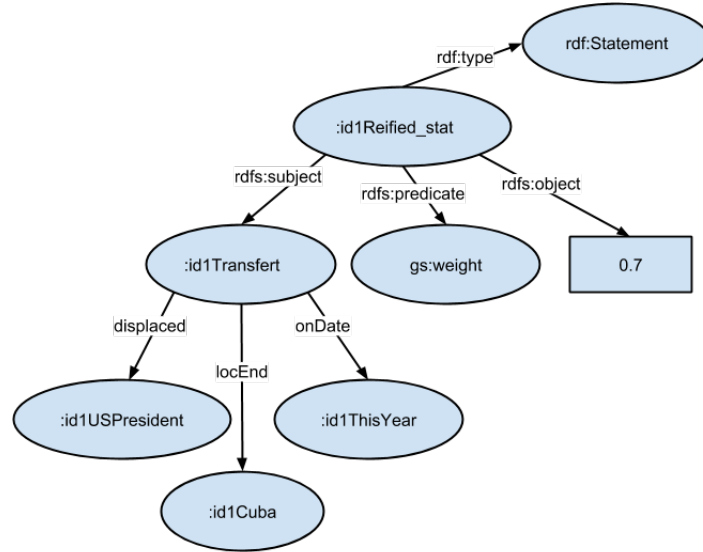


Fig. 1. RDF representation of uncertainty using reification

To deal with previous issues, we propose the *UncertaintyOntology* ontology which contains a concept (*Uncertainty*), a datatype property (*weight* which have *Uncertainty* as its domain and real values as range) and object properties (*isUncertain* and *hasUncertainProp* which respectively denote an uncertain individual (*Uncertainty* as domain and *owl:Thing* as range) and an uncertain property of a given individual (*Uncertainty* as Domain and *owl:Thing* as Range). This ontology can easily be integrated with our *geol* ontology or with any other ontology requiring some support for uncertainty.

This ontology (*UncertaintyOntology*) handles uncertainty occurring on each level of the triple. If the uncertainty concerns the resource, which denotes a subject or an object triple, so the property *isUncertain* is employed. If the triple's predicate is uncertain then we use *hasUncertainProp* to indicate the uncertainty. *UncertaintyOntology* is domain independent, it can be added to any other ontology since we assume that uncertainty occurs on each part of the sentence in a text.

To illustrate this representation, we provide in Figure 2, the RDF representation of Example 1's sentences. In the first sentence (on the left side), the uncertainty concerns the following triples :

- :id1Transfer, displaced, :id1USPresident.*
- :id1Transfer, locEnd, :id1Cuba.*
- :id1Transfer, onDate, :id1ThisYear.*

As we based on Bayesian approach, all these triples have an uncertainty of 0.7, expressed using the uncertainty marker *probably*. Whereas, in the second sentence, the uncertainty concerns only the property onDate, so, the triple *:id1Transfer, onDate, :id1ThisYear.* is uncertain.

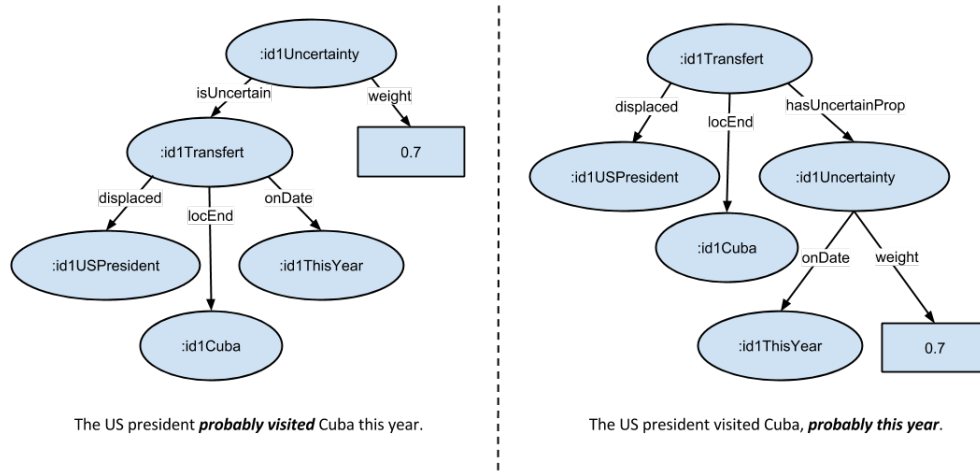


Fig. 2. RDF Knowledge representation of uncertainty in Example 1

Finally, we conclude that using this RDF representation, we identify three different cases of triple uncertainty. Figure 4 shows the representation of different patterns of uncertainty in RDF triples. Pattern 1 describes uncertainty on the object of the triple. Pattern 2 describes uncertainty on the subject and finally, pattern 3, uncertainty on the property.

This representation of uncertainty is more compact than reification and improves user understanding regarding the RDF graph.

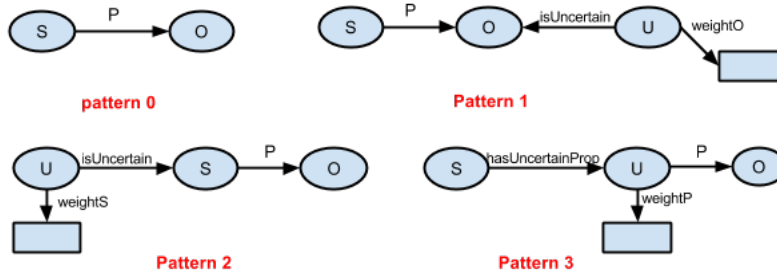


Fig. 3. RDF representation of Uncertainty patterns.

5 SPARQL Querying with uncertainty

The goal of our system is to enable end-users to query the extracted information. These queries take into account the presence of uncertainties by going through a rewriting. Our system discovers if such a rewriting is necessary by executing the following queries. First, we list all uncertain properties, using the query in Listing 1.1. The result is a set of triples (s,p,o) where p is an uncertain property.

```
PREFIX gs:<http://www.geolsemantics.com/onto#>
Select ?s ?prop ?o
Where {
    ?s gs:hasUncertainProp ?u.
    ?u gs:weight ?weight.
    ?u ?prop ?o.
}
```

Listing 1.1. SPARQL query Select uncertain properties

Then, we check if the predicates of each triple in the entry query appear in the result set. If so, we rewrite the query by adding the uncertainty on the given predicate using the pattern query in Listing 1.2. Finally, we inspect the query

```
PREFIX gs:<http://www.geolsemantics.com/onto#>
Select ?p ?weight
Where {...
    ?u gs:isUncertain ?p.
    ?u gs:weight ?weight.
...}
```

Listing 1.2. SPARQL query Select uncertain resources

result set of the rewritten query, in order to check if an uncertainty occurs on each resource (subject and/or object) extracted.

Furthermore, if a user wants to know the list of uncertainties in a given text, the query in Listing 1.3 is used to extract all uncertain data explicitly expressed. We consider that each linguistic extraction is represented according to the schema presented in Section 4. Our goal is now to provide a query interface to the end-user and to qualify the uncertainty associated to each query answer. Of course, the uncertain values that we are associating with the different distinguished variables of a query are directly emerging from the ones we are representing in our graph and which has been described in Section 4. Our system accepts any SPARQL 1.0 queries from the end-user. For testing reasons, we also have defined a set of relevant predefined queries, *e.g.*, the query in Example 3.

```

PREFIX gs:<http://www.geolsemantics.com/onto#>
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX v:<http://www.w3.org/2006/vcard/ns#>
SELECT distinct ?concept_uncertain ?obj ?weight
WHERE {
  {
    ?u a gs:Uncertainty.
    ?u gs:isUncertain ?concept_uncertain.
    ?u gs:weight ?weight
  }UNION {
    ?u2 a gs:Uncertainty.
    ?u2 gs:weight ?weight.
    ?s ?hasUncertainProp ?u2.
    ?u2 ?prop ?obj.}
}

```

Listing 1.3. SPARQL query : Select all uncertainties in the text

Example 3. Let us consider the query in Listing 1.4.

```

PREFIX gs:<http://www.geolsemantics.com/onto#>
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX v:<http://www.w3.org/2006/vcard/ns#>
Select ?date
Where {
  ?t gs:displaced ?p.
  ?p gs:role "president".
  ?t gs:locEnd ?l.
  ?l v:location-name "Cuba".
  ?t gs:onDate ?date.
}

```

Listing 1.4. SPARQL query : When did the president go to Cuba?

In order to make query submission easier for the end-user, we do not impose the definition of the triple patterns associated to uncertainty handling. Hence, the end-user just submits a SPARQL query without caring where the uncertainties are. Considering query processing, this implies to reformulate the query before its execution, *i.e.*, to complete the query such that its basic graph pattern is satisfiable in the face of triples using elements of our uncertain ontology.

We can easily understand that a naive reformulation implies a combinatorial explosion. This has direct impact on the efficiency of the query result set computation. This can be prevented by rapidly identifying the triple patterns of a query that are subject to some uncertainty. In fact, since our graphs can only represent uncertainty using one of the three patterns presented in Figure 4, we

```

PREFIX gs:<http://www.geolsemantics.com/onto#>
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX v:<http://www.w3.org/2006/vcard/ns#>
Select ?date ?w
Where {{
    ?t gs:displaced ?p.
    ?p gs:role "president".
    ?t gs:locEnd ?l.
    ?l v:location-name "Cuba".
    ?t gs:onDate ?date.
}UNION{
    ?t gs:displaced ?p.
    ?p gs:role "president".
    ?t gs:locEnd ?l.
    ?l v:location-name "Cuba".
    ?t gs;hasUncertainProp ?u.
    ?u gs:onDate ?date.
    ?u gs:weight ?w.
}}

```

Listing 1.5. Uncertainty query : When did the president go to Cuba?

can go through a pre-processing step that indexes these triples. To do so, we use a set of SPARQL queries (see Listing 1.1 and 1.2 which respectively retrieve the properties and subject with their weights). These values are stored in hash tables for fast access.

Therefore, Listing 1.5 corresponds to the rewriting of Listing 1.4. We introduced the uncertainty option and obtained the following results :

Sentence	Result	Uncertainty	Uncertainty Detail
(1)	?date = "20150101-20151231"	0.7	On the subject
(2)	?date = "20150101-20151231"	0.7	On the predicate

6 Conclusion and Perspectives

In this article, we addressed the quantification and qualification of uncertain and ambiguous information extracted from textual documents. Our approach is based on a collaboration between Natural Language Processing and Semantic Web technologies. The output of our different processing units takes the form of a compact RDF graph which can be queried with SPARQL queries and reasoned over using ontology based inferences. However, some issues are still unresolved, even for the linguistic community, such as: distinguish between deontic and epistemic meaning. Example: "He can practice sport." One can interpret this information as a permission and an other as an ability or a certainty.

This work mainly concerns the uncertainty expressed in the text, for future work we intend to consider the trust guaranteed to the source of the text. Indeed, the

source can influence the trustworthiness and the reliability of the declared information. Moreover, we plan to consider additional aspects of the information, such as polarity.

References

1. A. Auger and J. Roy. Expression of uncertainty in linguistic data. In *Information Fusion, 2008 11th International Conference on*, pages 1–8. IEEE, 2008.
2. F. Bobillo and U. Straccia. Fuzzy ontology representation using OWL 2. *International Journal of Approximate Reasoning*, 52(7):1073–1094, 2011.
3. T. A. Brown and E. H. Shuford. Quantifying uncertainty into numerical probabilities for the reporting of intelligence. Technical report, 1973.
4. P. Cimiano and J. Vlker. Text2Onto. In *Natural Language Processing and Information Systems*, volume 3513, pages 257–271. Springer Berlin, 2005.
5. W. W. W. Consortium et al. Rdf 1.1 semantics. 2014.
6. Z. Ding and Y. Peng. A probabilistic extension to ontology language OWL. In *System Sciences, 2004. Proceedings of the 37th Annual Hawaii international conference on*, pages 10–pp. IEEE, 2004.
7. Z. Ding, Y. Peng, and R. Pan. BayesOWL: Uncertainty modeling in semantic web ontologies. In *Soft Computing in Ontologies and Semantic Web*, pages 3–29. Springer, 2006.
8. M. J. Druzdzel. Verbal uncertainty expressions: Literature review. 1989.
9. D. Dubois and H. Prade. Formal representations of uncertainty. *Decision-Making Process: Concepts and Methods*, pages 85–156, 2009.
10. O. Hartig and B. Thompson. Foundations of an alternative approach to reification in RDF. *arXiv preprint arXiv:1406.3399*, 2014.
11. E. M. Johnson. Numerical encoding of qualitative expressions of uncertainty. Technical report, DTIC Document, 1973.
12. K. J. Laskey and K. B. Laskey. Uncertainty reasoning for the World Wide Web: Report on the URW3-XG incubator group. In *URSW*. Citeseer, 2008.
13. T. Lukasiewicz and U. Straccia. Managing uncertainty and vagueness in Description Logics for the Semantic Web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(4):291–308, 2008.
14. E. Marshman. Expressions of uncertainty in candidate knowledge-rich contexts: A comparison in english and french specialized texts. *Terminology*, 14(1):124–151, 2008.
15. B. McBride. Jena: Implementing the RDF model and syntax specification. In *SemWeb*, 2001.
16. A. Papafragou. Epistemic modality and truth conditions. *Lingua*, 116(10):1688–1702, 2006.
17. V. L. Rubin, N. Kando, and E. D. Liddy. Certainty categorization model. In *AAAI spring symposium: Exploring attitude and affect in text: Theories and applications, Stanford, CA*, 2004.
18. V. L. Rubin, E. D. Liddy, and N. Kando. Certainty identification in texts: Categorization model and manual tagging results. In *Computing attitude and affect in text: Theory and applications*, pages 61–76. Springer, 2006.
19. R. Sauri and J. Pustejovsky. From structure to interpretation: A double-layered annotation for event factuality. In *The Workshop Programme*, 2008.

20. G. Stoilos, G. B. Stamou, V. Tzouvaras, J. Z. Pan, and I. Horrocks. Fuzzy OWL: Uncertainty and the semantic web. In *OWLED*, 2005.
21. E. R. Watkins and D. A. Nicole. Named graphs as a mechanism for reasoning about provenance. In *Frontiers of WWW Research and Development-APWeb 2006*, pages 943–948. Springer, 2006.