Lexical Meaning Formal Representations Enhancing Lexicons and Associated Ontologies

Maria Gritz

Herzen State Pedagogical University of Russia,
Faculty of Philology
maria.gritz@yandex.ru

Abstract. The paper represents an outline of a new technique aimed at improvement of lexicon-to-ontology mapping technology. The technique is integrated within the Ontolex-Lemon lexicon model by supplying a lexical concept of an ontology class with a description logic based formal definition. A natural language definition used to describe the lexical concept is transformed into a DL-definition; the resulting DL-definition is associated with a graph-like join of domain ontology properties. As a result, a related lexical unit is mapped to several bound ontology units rather than to a single ontology class. DL-based and graph-based lexical meaning formal representations are applied for lexical sense disambiguation within a lexicon and for extension of class and property taxonomies of an associated ontology.

Keywords: lexicon model, domain ontology, lexical meaning, formal definition, Ontolex-Lemon, OWL 2 DL, SROIQ(D).

1 Introduction

Due to extensive growth of amount and variety of World Wide Web content, the task of relevant information retrieval is becoming more and more challenging. Traditional keyword-based search engines show high recall but low precision since the search technology does not provide any formal account of semantics of keywords [14, 16]. The Semantic Web initiative has been launched at the turn of the XXI century in pursuit of augmentation of Web search technologies with applications able to conduct knowledge-based analysis of meaning conveyed by natural language expressions [23]. The semantic-based Web search is supposed to render accurate search results either by providing the requested data retrieved from knowledge bases or by giving out a list of relevant documents retrieved from a Web document collection.

Ontologies are supposed to be the key to an advanced search technology, providing formal specifications of vocabulary units used to represent a domain [9]. The scope of vocabulary units includes individuals representing single entities of a domain; classes rendering subsets of domain entities; object properties introducing binary relations on a domain; and datatype properties, which assign literal and numerical characteristics to subsets of domain entities through specific datatypes. Classes and properties are introduced into class and property taxonomies of an ontology. Ontologies are subjected to population by instantiating classes with individuals and assigning values to properties. Within a semantic search engine, assertions are included in the body of an ontology [16] or are used to form a correlated knowledge base implemented to describe particular states of affairs [23].

Numerous ontology-based search engines have been developed during the recent years in order to improve the precision of traditional keyword-based search: MIRO [17], IBRI-CASONTO [18], and Fuzzy semantic search engine [16] are just few of them. The list of ontology-based search engines includes a variety of applications. Some of them function as question answering systems conducting semantic search over ontologies and correlated knowledge bases to provide an exact answer to a question. Pythia [22], Wolfram alpha[1], and Kngine[2] are suitable examples. Google[3] extends a traditional keyword-based search engine with functions of a question answering system. In other words, apart from retrieving a list of Web documents that contain the requested information, the search engine provides the required data stored in a knowledge base. MIRO [17] and Fuzzy semantic search engine [16] conduct ontology-based semantic search to return lists of relevant documents.

Within the process of ontology-based semantic search, users' queries and Web documents acquire ontologically motivated semantic representations [7]. Web documents are annotated either with classes of an ontology [17] or with assertions [6]. Lexical units forming a user's query are mapped to units of an ontology [21]. The mapping is done syntactically, i.e. by virtue of syntactic similarity measurement [21], or semantically, i.e. by using data stored in a lexicon [22].

The major bottleneck in ontology-based semantic search development lies in designing an efficient technology of semantic analysis of users' queries, which provides accurate matching of lexical units with units of an ontology. The vast variety of lexical means of expression represented by natural languages along with highly developed homonymy, synonymy, and polysemy within lexical systems result in the exemplified use of ontology-based search engines being limited to a particular domain, for example, a soccer domain [23] or a book domain [16].

The rest of the paper is organized as follows: in sections 2 and 3 we discuss lexical analysis implementation in operation of natural language interfaces to ontologies and lexicon models applied during the analysis. Section 4 is devoted to basic drawbacks of the current lexicon-to-ontology mapping technology. Section 5 introduces our approach towards associating lexicon units with units of an ontology. Section 6 represents the results of experimental implementation of the novel technique. Section 7 provides some brief conclusions as well as a vision of perspectives and future work.

[1] http://m.wolframalpha.com
[2] http://www.kngine.com
[3] https://www.google.com

2 Natural Language Interfaces to Ontologies

Natural language interfaces (NLI) are developed to conduct semantic analysis and to provide a formal representation of a query's semantics [20]. Semantic analysis is executed by mapping natural language phrase structures to units of an ontology. A formal representation of a query's semantics must be given in a form of a string with non-logical symbols corresponding to units of an ontology involved in the analysis. The string must be written in a language such as SPARQL [19], which is understandable for an agent conducting semantic search over ontologies and correlated knowledge bases.

NLIs typically involve an extensible lexicon, which is at least partly generated automatically from an ontology, in the process of semantic analysis of a query. The involvement of linguistic data on lexemes provided by a lexicon in the process of semantic analysis of a query differs from one NLI system to another. The NLI of FREyA [5] and PANTO [25] conduct syntactic analysis of a query by virtue of the Stanford Parser and use a lexicon to link parse tree nodes to ontology units through synsets introducing synonyms and spelling variants of lexemes representing ontology units.

The NLIs of ORAKEL [3] and Pythia [22] harness information stored in a lexicon to conduct syntactic analysis of a query and to obtain an ontology-based formal representation of its semantics. The linguistic data on closed class words: determiners, conjunctions, interrogative pronouns, prepositions is stored in a domain-independent part of a lexicon. The current paper is focused on development and implementation of a domain-specific part of a lexicon, which supplies linguistic data on open class words: nouns, verbs, and adjectives.

2.1 NLI of ORAKEL

The ORAKEL parser [3] processes a query by mapping tokens to units of a lexicon, which is developed in accordance with the LexOnto model [4]. Units of a lexicon are introduced by virtue of elementary tree families uniting tree-like syntactic representations of a lexeme. A family of elementary trees renders information on syntactic categories of a lexeme and its arguments together with grammatical and lexical constraints imposed on them. Grammatical constraints for arguments are introduced by genus and head feature values. Lexical constraints are defined as ontological restrictions: a lexeme in an argument position should be mapped to an ontology class or a data value range which is either equivalent or subsumed by a domain or a range of an ontology property the head of an elementary tree is mapped to.

Elementary trees are combined by the parser to produce a parse tree. The head nodes of the tree representing nouns, verbs, and adjectives are provided semantics by virtue of lambda expressions describing an unary or a binary predicate or a constant which corresponds to a unit of an underlying ontology. The Query Interpreter of the ORAKEL system combines semantic representations of tree nodes to produce a formal representation of a query's semantics. The formal representation is provided in the form of a FOL-like formula augmented with query, count, and arithmetic operators, which is transformed into a SPARQL query.

2.2 NLI of Pythia

The NLI of Pythia [22] produces syntactic and semantic analysis of a user's query in parallel using information stored in a lexicon. An entry of the lexicon is organized in accordance with the LexInfo model [2], which is used to provide syntactic and semantic data on an ontology unit's lexicalization. An ontology unit, which is an individual, a property, or a class of an ontology, is provided semantic representation by virtue of a DUDE (Dependency-based Underspecified Discourse Representation Structure) [1]. A DUDE includes a name of a predicate corresponding to an ontology unit and identifiers of its arguments' positions in a family of elementary LTAG trees. The LTAG trees expose a number of possible syntactic representations of a lexeme, which is mapped to the ontology unit.

Elementary LTAG trees representing lexical units of a query are used by the inbuilt parser to obtain an LTAG-derivation tree, and entries of a lexicon provide enough data to build a parse tree by implementing substitution and adjoin operations. DUDEs are merged to form a Discourse Representation Structure (DRS), the obtained DRS of a user's query is converted into a SPARQL query.

3 NLI Integrated Lexicon Models: LexInfo and LexOnto

LexInfo [2] and LexOnto [4] are OWL-based lexicon models used to link units of an ontology to units of a lexicon thereby providing morphological, syntactic, and semantic data on lexemes used to express ontology units. Meanwhile, lexemes acquire sense being associated with ontology units. For instance, verbs are supposed to represent properties of an ontology, whereas common nouns are typically associated with domain ontology classes. In the LexInfo model compound words are subjected to decomposition with every part being associated with an ontology unit.

Both lexicon models are intended chiefly to provide for a domain-specific ontology lexicon. The core difference between LexInfo and LexOnto lies in the mode of lexicon-to-ontology units mapping. In LexOnto subcategorization frames introducing arguments attached by verbs and relational nouns are associated with single or joined properties of a domain ontology. In LexInfo verbs and relational nouns are mapped to object and datatype properties in a straightforward fashion, whereas their syntactic behavior is specified by virtue of specific subcategorization frames with arguments being mapped to domains and ranges of the properties associated with the lexemes. In both frameworks all lexemes acquire lemmas and a list of form variants distinguished by mood, gender, number, case, degree, person, etc.

LexInfo and LexOnto specify syntactic behavior of a verb or a relational noun through a verbal or a nominal subcategorization frame accordingly. A verbal subcategorization frame encodes number and sort of a verb's arguments: a subject, a direct object, and optionally one or two prepositional complements for a transitive verb; a subject and a prepositional complement for an intransitive verb. A nominal subcategorization frame includes one or two prepositional complements, and an argument position titled as *external subject* is filled whenever a relational noun is used in the role of a predicate, which is related to a subject by virtue of copula. A subject is included into nominal subcategorization frames within the LexInfo system, but it is stored separately by LexOnto. Apart from that, LexOnto provides subcategorization frames for participles attaching a prepositional complement. A binary frame is mapped either to a single property or to a 2 x 2-Join of properties; a ternary frame is mapped either to a 3 x 2-Join or to a 2 x 2-Join' of properties whenever a joined position is mapped to an argument of the frame; a quaternary frame is associated with a 4 x 2-Join or with a 3 x 2-Join' of ontology properties.

Adjectives used as adjectival modifiers of a noun are supposed to subcategorize for a modified noun. An adjective and a modified noun are mapped to ontology classes undergoing intersection whenever the adjective is associated with a particular class entering a class taxonomy of an ontology. Adjectives of this kind are referred to as intersective or class adjectives and are successfully handled by LexInfo. Subjective adjectives, on the other hand, do not have fixed extensions on a domain and their interpretation is context dependent. An adjective of this kind is mapped to a property of an ontology with special constraints being imposed on the property's range, whereas a modified noun is associated with a subclass of the domain of the property.

A lexicon-to-ontology mapping scheme depends on a kind of a property which is associated with an adjective. LexInfo maps literal adjectives such as *blue* or *skillful* to object properties. Evidently, an ontology-based interpretation of the adjective *skillful* has to be different if it subcategorizes for the noun *gardener* or for the noun *surgeon*. Yet, the value constrains that have to be imposed on a domain of an object property a literal adjective is mapped to are not proposed.

Within the frameworks of LexInfo and LexOnto, a scalar adjective like *long* or *big* is mapped to a datatype property on occasion particular constraints are imposed on the data value range. Positive or negative polarity has to be set for this data value range to indicate if the value has to increase or to decline to give an appropriate formal account of comparative and superlative forms of the adjective. One should bear in mind that the constraints might be regionally or culturally specific and take into account the units of measurement. LexInfo also proposes value constraints imposed on a datatype property domain to model a scalar adjective's semantics since data value constraints are supposed to be different, for instance, when the noun *man* or the noun *woman* is modified by the adjective *tall*.

4 An Outline of the Fallacies of Lexicon-to-ontology Mapping Technology

Within the framework of Natural Language Interfaces to ontologies, lexemes acquire semantics with reference to ontology units. Whenever an NLI integrates a lexicon model instantiated by LexInfo and LexOnto, word sense disambiguation is done by virtue of semantic analysis of a target lexeme's syntactic behavior. On occasion the syntactically bound lexemes of a target lexeme respect the ontological restrictions imposed through lexicalization of an ontology unit, the target lexeme is supposed to refer to the ontology unit. These restrictions expect lexical fillings of slots in lexemes' subcategorization frames to be coreferential on a domain with particular class names or data value ranges that are predefined by one-to-one correspondence between syntactic role slots and domain/range constituting classes/data value ranges. This correspondence is set within the process of lexicalization and is scripted by virtue of elementary trees with ontological restrictions being imposed on particular nodes that acquire syntactic roles defined in a subcategorization frame.

The technology of meaning acquisition and disambiguation provided by lexicon-to-ontology mapping frameworks poses a high demand on taxonomy organization and entity coverage of a lexicon. In other words, all classes and properties that can possibly be lexicalized by a user in a query should be included in taxonomies of a domain ontology, and all possible contexts of lexemes' use should be taken into account in the process of ontology units' lexicalization. These demands appear to be virtually unfeasible, yet, their necessity is easily illustrated by the following examples of problematic cases of lexicon-to-ontology mapping.

An intransitive verb *pass*, which subcategorizes for a subject and a prepositional complement attached by the preposition *through*, is provided as an instance of an ambiguous verb by ORAKEL developers [3]. The ambiguity is resolved by mapping an argument playing the role of a subject to a subclass of the ontology class *River* or to a subclass of the class *Highway*. Since the ontology classes *River* and *Highway* are disjoint, whenever a lexeme denoting a river or a kind of a river attains the role of a subject, the predicate *pass* is mapped to the object property *flow_through* with the class *River* as domain and the class *City* as range. Whenever a lexeme denoting a highway or a kind of a highway attains the role of a subject, the predicate *pass* is mapped to the object property *located_at_highway* with the class *City* as domain and the class *Highway* as range. If the class *River* is used to designate all kinds of waterways in a geographical object domain, the class should subsume the classes labeled as *Creek* or *Channel*, for instance, so that ontological restrictions could be respected for a variety of contexts. Moreover, the semantic analysis of the queries concerning pipelines or railways, for instance, will fail if the properties *flow_through (River, City)* and *located_at_highway (City, Highway)* are the only options to choose from.

Ontological restrictions imposed on a verb's arguments by a join of associated properties should also be subjected to thorough reification. For instance, an alternative OntoSem lexicon model [13] defines several senses of the transitive verb *to address*, one of them being described as *to talk to* and exemplified with the sentence *He addressed the crowd*. The verb *to address* used in that sense is associated with the 2 x 2-Join of object properties *hasAgent (SpeechAct, Human)* and *hasBeneficiary (SpeechAct, Human)*. Yet, even the illustrating example shows the necessity of enhancing the constraints for verbalization of the direct object.

The authors of LexOnto [4] and ORAKEL [3] exemplify scalar adjectives' interpretation by mapping the adjective *big* subcategorizing for the noun *city* to the datatype property *inhabitants (City, xsd:integer)* and specifying the threshold number of city inhabitants which is required to evaluate a city as big. However, a user giving a request for a list of big cities could be interested in most densely populated cities or in cities occupying the largest areas. In these cases, a correct answer to the query could be given only if an NLI system maps the adjective *big* to the datatype properties *populationDensity (City, PopulationDensity)* or *area (City, Area)* accordingly.

5 Lexical Meaning Formal Representations Integrated within the Ontolex-Lemon Model

In order to resolve the issues of semantic ambiguity and inaccuracy of ontology-based lexical semantics representations, we propose to enhance the technology of lexicon-to-ontology units mapping by introducing description logic based formal definitions of ontology class representing lexemes in the scope of semantic data provided by an ontology lexicon. These definitions are intended to provide a formal account of a lexeme's meaning. In the current research the notion of lexical meaning is equated to intension, which is understood as '*a function from a set of possible worlds to a set of all subsets of homogeneous n-ary relations on a domain:* $Int_V: W \rightarrow 2^{D^n}$' [8]. Whenever an informal notation is preferred, an intension should be defined as a scope of indispensable attributes that a referent of a lexeme has to possess on a domain. A DL-definition is obtained by virtue of natural language definition transformation conducted in accordance with a set of transformation rules described by Völker et al. [24] and by Gritz [8][4]. NL-definitions are retrieved from lexicons, in which they are used to provide informal descriptions of intensions shared by lemmas united in synsets as synonyms or spelling variants.

Resulting DL-definitions are presumed to be introduced by virtue of the OntoLex-Lemon model developed to represent data on ontology units' lexicalizations [11, 12]. Within the framework of the OntoLex-Lemon model, lexical concepts are associated with particular ontology units by virtue of the *isConceptof* property and its inverse property *concept*. A lexical entry, which represents a word, an affix, or a multiword expression, evokes one or more lexical concepts endowed with NL-definitions. Simultaneously, a lexical entry is bound with one or more lexical senses with optional restrictions on register, domain, or context being introduced. Each lexical sense refers to one ontology unit by virtue of the functional *reference* property. Each lexical sense is associated with a lexical concept by means of the *isLexicalizedSenseOf* property and its inverse property *LexicalizedSense*.

Figure 1 provides an instance of a lexical entry representing the noun *coach*, which evokes two lexical concepts defined by Open Multilingual Wordnet 2.0 (OMW)[5]: *private instructor* and *manager*. *Private instructor*, a concept of the DBpedia[6] class *Coach*, is associated with the first sense of the lexeme *coach*, which could be defined as *a person who gives private instruction*. *Manager*, a concept of the DBpedia class

[4] Please note that the set of transformation rules has been augmented with the solutions for formalization of scalar and literal adjectives proposed within LexOnto and LexInfo systems (see Section 3).
[5] http://compling.hss.ntu.edu.sg/iliomw/omw
[6] http://mappings.dbpedia.org/server/ontology

SportsManager, is associated with the second sense of the lexeme *coach*, which could be defined as *someone in charge of training an athlete or a team*. Yet, the class *Coach* might be related to both concepts, which results in semantic ambiguity for the second sense of the lexeme *coach*. The lexical sense acquires reference to the ontology classes *Coach* and *SportsManager* despite the fact that *reference* is a functional property.

In order to overcome the ambiguity, the NL-definitions used to characterize corresponding lexical concepts have been formalized by virtue of NL-DL definition transformation. Concepts and roles forming complex descriptions in DL-definitions have been associated with units of the DBpedia ontology. As a result, the lexeme *coach* has acquired meaning by being mapped to joins of ontology units rather than to a single class of the ontology. The joins form graph structures with properties corresponding to edges, classes and data value ranges corresponding to vertices, therefore the joins are referred to as graph-definitions representing lexical meanings of associated lexemes.

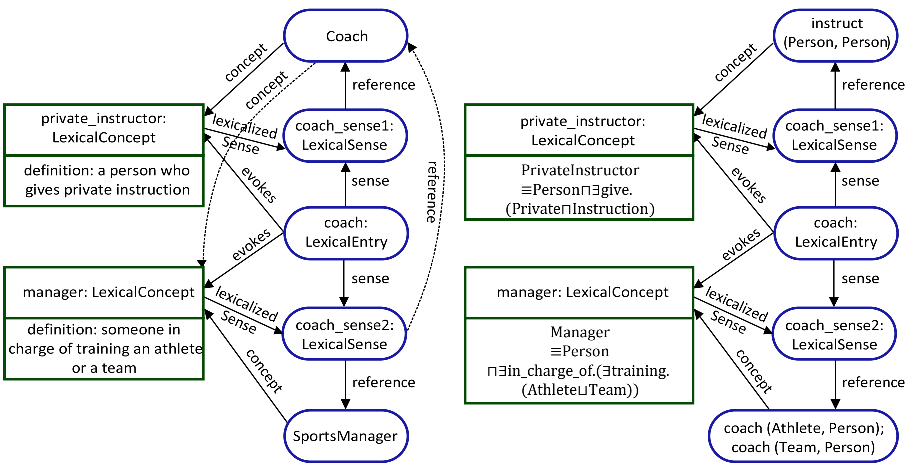

Fig. 1. Semantic disambiguation of the noun *coach* within the Ontolex-Lemon model

6 An Application of DL-definitions and Graph-definitions in Lexicon-to-ontology Mapping

A DL-definition is a terminological axiom stating concept equivalence, which binds an atomic concept with a complex description obtained by means of specific concept constructors. In the current research the constructors presumed by SROIQ(D) syntax [10], which is an OWL 2 DL [15] compatible description logic, are applied. Complex descriptions are chains of intersections between atomic concepts and complex descriptions obtained by posing universal, existential, or number restrictions on a role's range and by virtue of Boolean constructors: conjunction, disjunction, and negation.

Atomic concepts, roles, or their combinations introduced in different joints of a chain of intersections are associated with subclasses of domains and ranges of object and datatype properties of an ontology. The ontology properties form a graph-definition associated with a lexical concept of an ontology class. DL-based and graph-based definitions related to the OMW lexical concept *archeologist* are given in the Figure 2 for the purpose of illustration. All lexical senses associated with the lexical concept *archeologist* are supposed to refer to the interconnected properties forming a graph rather than to the DBpedia class *Archeologist*. The resulting graph-definition specifies the ontological restrictions for syntactic arguments of the verbs associated with the following DBpedia object properties: *activity (Person, Thing), focus (Thing, Thing), related (Thing, Thing), era (Thing, Thing)*. The constraints are supposed to be applied within discourse on human occupations.

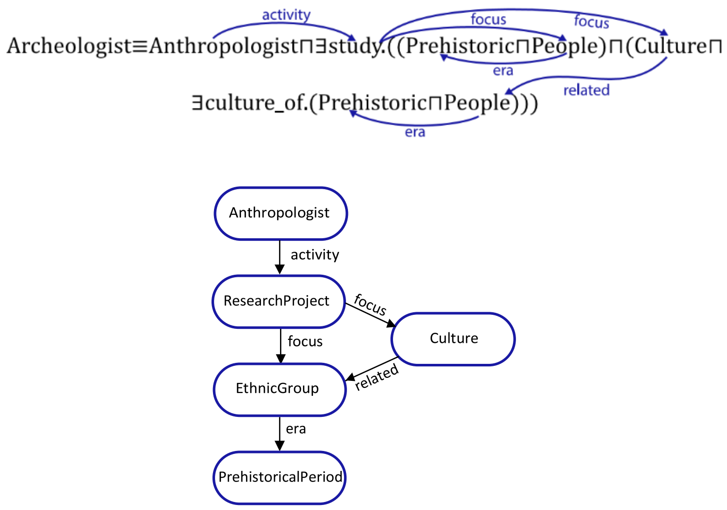

Fig. 2. Mapping a DL-definition to a graph-definition associated with the lexical concept *archeologist*

This kind of specification appears to be the key to resolution of problematic cases of mapping verbs to object properties of an ontology, which are exemplified in the Section 4. Let us presume that DBpedia contains the object properties: *pass_through (Thing, Place), establish (Agent, Thing),* and *hasBeneficiary (SpeechAct, Thing)*. In order to specify an ontological restriction for a lexeme associated with the domain of the object property *pass_through (Thing, Place)*, one should produce a DL-definition bound with the OMW lexical concept *itinerary* associated with the domain of the property. The DL-definition:

$$Itinerary \equiv \exists established.(Line \sqcap \exists line\_of.(Travel \sqcup Access)), \qquad (1)$$

acquires the following graph-based interpretation within the framework of DBpedia: *establish (Agent, RouteofTransportation)*; *pass_through (RouteofTransportation, Place)*. With the ontological restriction being specified as *pass_through*

(RouteofTransportation, Place), the lexical units associated with the domain of the object property *pass_through (Thing, Place)* are supposed to be coreferential with subclasses of the DBpedia class *RouteofTransportation*, i.e. with the classes: *Bridge, RailwayLine, RailwayTunnel, Road, RoadJunction, RoadTunnel, WaterwayTunnel*.

In order to impose an ontological restriction on a verb argument which is associated with the range of the object property *hasBeneficiary (SpeechAct, Thing)*, a DL-definition associated with the OMW lexical concept *addressee* was mapped to the object property *hasBeneficiary (SpeechAct, Agent)*. As a result, the DBpedia class *Agent* was set as an ontological restriction for a syntactic argument associated with the range of the property *hasBeneficiary (SpeechAct, Thing)*.

The proposed attitude might be extended to improve semantic representations of adjectives. With a view to define the properties an adjective is mapped to, one has to produce a DL-definition associated with a lexical concept evoked by a modified noun. For instance, whenever the adjective *big* modifies the noun *city*, the adjective should be associated with the DBpedia datatype properties *area (City, Area)* and *populationDensity (City, PopulationDensity)*. The reason is that these datatype properties compose a graph-definition bound with the OMW lexical concept *urban center*, which is evoked by the noun *city*.

In order to provide a brief description of advantages and shortcomings of the proposed approach to lexicon-to-ontology mapping, we have retrieved 50 lexical concepts from Open Multilingual Wordnet 2.0, the concepts that should be associated with subclasses of the DBpedia class *Person*. The NL-definitions used to characterize intensions of the lexemes associated with the lexical concepts were transformed into DL-based formal definitions.

All DL-definitions used to characterize 50 retrieved lexical concepts associated with 39 DBpedia classes were successfully linked to graph-definitions. Graph-definitions were the keys to resolution of 11 cases of semantic ambiguity, which arose every time two or more lexical concepts were defined as concepts of the same ontology class. The list of these classes includes the classes: *Coach*, *Judge*, *Politician*, *Referee*, and *Ambassador* among others. The set of graph-definitions provides information on the ontological restrictions that should be imposed on arguments of the lexemes associated with the object properties: *profession (Person, Thing)*, *education (Person, Thing)*, *activity (Person, Thing)*, *management (Thing, Thing)*, *specialization (Thing, Thing)*, *created (Person, Work)*. For instance, within discourse on human occupations the arguments associated with the domain of the object property *management (Thing, Thing)* should be restricted to the ones mapped to subclasses of the DBpedia classes *Person* and *WrittenWork*. The arguments mapped to the range should be restricted to the ones associated with subclasses of the DBpedia classes: *Person, Activity*, *Place*, and *MeanOfTransportation*. The arguments associated with the domain of the object property *specialization (Thing, Thing)* should be restricted to the ones mapped to subclasses of the DBpedia class *Person,* whereas the lexical units associated with the range of the property are supposed to be coreferential with subclasses of the class *Science*.

In 78% of cases newly proposed classes and properties had to be used in order to compile suitable graph-based definitions associated with particular lexical concepts.

Overall, 43 classes and 9 object properties have been proposed to enhance the class and property taxonomies of DBpedia. For instance, the class *Anthropologist* and the class *Culture* have been proposed in order to develop the graph-definition represented in the Figure 2. Some of the proposed properties appear to be crucial for formal description of human occupation subdomain: *instruct (Person, Person), solve (Thing, Thing), represent (Thing, Thing), find (Thing, Thing)*. Two object properties that have been proposed: *appoint (Agent, Person)* and *appointed (Person, Thing)*, form a 2 x 2 Join' which is supposed to be mapped to the subcategorization frame of the verb *to appoint* containing three arguments. 13,5 % of the proposed ontology units were used more than once for the purpose of graph-based representation of lexical concepts of human occupations. 6% of proposed units subsume at least one suggested unit. For instance, the introduced class *Science* possesses 6 subclasses that were suggested to represent different areas of research: *Economics, Egyptology, Linguistics, Philosophy, Psychology, History*. Consequently, the extension of an ontology's taxonomies in the process of graph-definitions formation should be considered domain oriented and therefore fruitful.

Simultaneously, a soft spot of the proposed technique arises. Units composing DL-based and graph-based definitions acquire an irregular match that might complicate the technique implementation in lexicon modelling (see Table 1). For instance, the range of the object property *education (Person, Thing)* is associated with the atomic concept *Psychology* within the DL-definition related to the lexical concept *psychologist*. The same object property gets its range mapped to the atomic role *travel_in*, which is in turn associated with the domain of the property *equipment (Activity, Thing)* used within the graph-definition related to the lexical concept *spaceman*. Finally, the range of the object property *education (Person, Thing)* happens to be related to the complex description ∃*compete_in.Sports*, which in turn is used to define an existential restriction imposed on the atomic role *trained_to,* within the DL-definition related to the lexical concept *athlete*. The range of the object property *created (Person, Work)* is associated with an intersection of the concepts *Creative* and *Work* within the DL-definition related to the lexical concept *artist*. Within the same DL-definition the range of the object property *picture (Thing, Thing)* is mapped to the atomic concepts *Sensitivity* and *Imagination* that compose an intersection used to characterize an existential restriction imposed on the atomic role *show*. The case of the object property *activity (Person, Thing)* being mapped to the atomic concept *Scientist* within the DL-definition associated with the lexical concept *psychologist* should also be taken into consideration.

These examples reveal the necessity of making ad-hoc decisions in order to link units of DL-based definitions of lexicon units' semantics with units of an associated ontology. Hence, even though the proposed technique seems to be an appropriate tool for revision and improvement of lexicon-to-ontology mappings conducted in relevance to a particular domain of discourse, the large-scale implementation of the technique is yet to be achieved.

DL-definition	graph-definition
$Artist \equiv Person \sqcap \exists produce.((Creative \sqcap Work) \sqcap \exists show.(Sensitivity \sqcap Imagination))$	created (Person, ArtWork); picture (ArtWork, Sensitivity); picture (ArtWork, Imagination)
$Athlete \equiv Person \sqcap \sqcap \exists trained\_to.(\exists compete\_in.Sports)$	education (Person, Contest)
$Psychologist \equiv Scientist \sqcap \exists trained\_in.Psychology$	activity (Person, Science); education (Person, Psychology)
$Spaceman \equiv Person \sqcap \exists trained\_to.(\exists travel\_in.Spacecraft)$	education (Person, Management); equipment (Management, Spacecraft)

Table 1. Examples of description logic based and graph-based notations of lexical semantics

7 Conclusion

The brief study introduced in the current paper has shown that DL-based and graph-based formal specifications of a lexeme's meaning improve the accuracy of lexicon-to-ontology mappings by resolving cases of semantic ambiguity among ontology class representing lexemes. At the same time, the semantics of property representing lexemes is subjected to reification through specification of ontological restrictions imposed on arguments entering the lexemes' subcategorization frames.

The novel technique of ontology lexicon modelling is applicable under the condition of an ontology's taxonomies being limited and allows to deny the impracticable demand for summarization of all possible grammatical contexts of a lexeme's use. Meanwhile, the process of graph-definitions formation stimulates the development of class and property taxonomies of a domain ontology. However, a regular correspondence between units composing DL-based and graph-based definitions is yet to be found. A set of rules associating atomic roles, atomic concepts, and complex descriptions of a DL-definition with classes and data value ranges representing domains and ranges of ontology properties has to be introduced in order to make the technique applicable in a large-scale fashion.

References

1. Cimiano, P.: Flexible semantic composition with DUDES. In: Proceedings of the 8th International Conference on Computational Semantics, pp. 272–276. The Netherlands, Tilburg (2009)
2. Cimiano, P., Buitelaar, P., McCrae, J., Sintek, M.: LexInfo: A declarative model for the lexicon-ontology interface. In: Web Semantics: Science, Services and Agents on the World Wide Web 9(1), 29–51 (2011)
3. Cimiano, P., Haase, P., Heizmann, J., Mantel, M.: Orakel: A portable natural language interface to knowledge bases (2007). https://pub.uni-bielefeld.de/download/2497310/2525113, last accessed 2018/08/18

4. Cimiano, P., Haase, P., Herold, M., Mantel, M., Buitelaar, P.: LexOnto: A model for ontology lexicons for ontology-based NLP. In: Proceedings of the OntoLex07 Workshop held in conjunction with ISWC'07 (2007)
5. Damljanovic, D., Agatonovic, M., Cunningham, H.: Natural language interfaces to ontologies: Combining syntactic analysis and ontology-based lookup through the user interaction. In: Extended Semantic Web Conference, pp. 106–120. Springer, Berlin, Heidelberg (2010)
6. Fazzinga, B., Gianforme, G., Gottlob, G., Lukasiewicz, T.: Semantic Web search based on ontological conjunctive queries. Web Semantics: Science, Services and Agents on the World Wide Web 9(4), 453–473 (2011)
7. Fazzinga, B., Lukasiewicz, T.: Semantic search on the Web. Semantic Web 1(1, 2), 89–96 (2010)
8. Gritz, M.: Towards lexical meaning formal representation by virtue of the NL-DL definition transformation method. In: Proceedings of the 3rd International Conference on Computational Linguistics in Bulgaria, pp. 23–33. The Institute for Bulgarian Language Bulgarian Academy of Sciences, Sofia (2018)
9. Guarino, N.: Formal ontology and information systems. In: Guarino, N. (ed.) Formal Ontology in Information Systems. Proceedings of the FOIS'98, pp. 3–15. IOS Press, Amsterdam (1998)
10. Horrocks, I., Kutz, O., Sattler, U.: The even more irresistible *SROIQ*. In: Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning, pp. 57–67. AAAI Press (2006)
11. Lexicon model for ontologies: Community Report. In: Cimiano P., McCrae J. P., Buitelaar P. (eds.) Final Community Group Report (2016). https://www.w3.org/2016/05/ontolex, last accessed 2018/10/10
12. McCrae, J., Bosque-Gil, J., Gracia, J., Buitelaar, P., Cimiano, P.: The OntoLex-Lemon model: development and applications. In: Proceedings of eLex 2017 conference, pp. 19–21. (2017)
13. McShane, M., Nirenburg, S., Beale, S.: Language understanding with ontological semantics. Advances in Cognitive Systems 4, 35–55 (2016)
14. Mehta, A., Zatakia, S., Deulkar, K.: Comparative study of web search methods using ontology. International Journal of Computer Science and Information Technologies 6(6), 5497–5499 (2015)
15. OWL 2 Web Ontology Language Primer. In: Hitzler, P., Krötzsch, M., Parsia, B., Patel-Schneider, P. F., and Rudolph, S. (eds.) W3C Recommendation. 2nd edn. (2012). https://www.w3.org/TR/owl2-primer, last accessed 2018/10/10
16. Remi, S., Varghese, S. C.: Domain ontology driven fuzzy semantic information retrieval. Procedia Computer Science 46, 676–681 (2015)
17. Salam, F.: New semantic indexing and search system based on ontology. In: 4th International Conference on Emerging Intelligent Data and Web Technologies, pp. 313–318. IEEE (2013)
18. Sayed, A., Al Muqrishi, A.: IBRI-CASONTO: Ontology-based semantic search engine. Egyptian Informatics Journal 18(3), 181–192 (2017)
19. SPARQL 1.1 Overview. W3C Recommendation (2013). https://www.w3.org/TR/sparql11-overview, last accessed 2018/10/10
20. Šukys, A., Nemuraitė, L., Butkienė, R.: SBVR based natural language interface to ontologies. Information Technology and Control 46(1), 118–137 (2017)

21. Tran, T., Cimiano, P., Rudolph, S., Studer, R.: Ontology-based interpretation of keywords for semantic search. In: Proceedings of the 6th International Semantic Web Conference (ISWC'07), pp. 523–536. Springer, Berlin, Heidelberg (2007)
22. Unger, C., Cimiano, P.: Pythia: Compositional meaning construction for ontology-based question answering on the semantic web. In: International Conference on Application of Natural Language to Information Systems, pp. 153–160. Springer, Berlin, Heidelberg (2011)
23. Vanjulavalli, N., Kovalan, A.: Ontology based semantic search engine. International Journal of Computer Science Engineering & Technology 2(8), 1349–1353 (2012)
24. Völker, J., Haase P., and Hitzler, P.: Learning expressive ontologies. In: Buitelaar, P., Cimiano, P. (eds.) Ontology Learning and Population: Bridging the Gap between Text and Knowledge, pp. 45–69. IOS Press, Amsterdam (2008)
25. Wang, C., Xiong, M., Zhou, Q., Yu, Y.: Panto: A portable natural language interface to ontologies. In: European Semantic Web Conference, pp. 473–487. Springer, Berlin, Heidelberg (2007)