Chapter 6
Unnatural Language Processing: Characterizing the Challenges in Translating Natural Language Semantics into Ontology Semantics

Kent D. Bimson and Richard D. Hull

6.1 Introduction

In recent years, significant work has been performed on using ontologies as a basis for extracting knowledge from text and representing it as ontology knowledge structures, typically using Web Ontology Language (OWL; Bechhofer et al. 2004) or a Resource Description Framework (RDF) (RDF—Semantic Web Standards 2004) triple store. There are many good reasons for doing so, such as to capture specific knowledge from text that is needed for a domain analysis problem, to structure and normalize semantics for machine interpretation, to improve semantic search, or to share information via the Semantic Web, to name only a few.

Significant challenges arise, however, when ontology designers attempt to model the richness of natural language (NL) semantics using constrained ontology semantics (Bimson 2009), a process that leads to what we call unnatural language processing, or attempting to model the richness of NL semantics using constrained ontology semantics. Unnatural language processing is a consequence of the fact that the semantics natural to languages is not so easily or naturally represented within standard ontology representation languages, at least not without significant effort. As a result, much of the semantic content in NL text is lost in translation to RDF or OWL. Semantic extensions added to OWL 2 (OWL 2 Web Ontology Language 2012), including richer data types and data ranges, qualified cardinality restrictions,

K. D. Bimson (✉) · R. D. Hull
Intelligent Software Solutions, Inc., 5450 Tech Center Drive,
Suite 400, Colorado Springs, CO 80919, USA
e-mail: kent.bimson@issinc.com

K. D. Bimson
Department of Electrical & Computer Engineering, The University of New Mexico,
Albuquerque, NM, 87131-0001, USA

R. D. Hull
e-mail: richard.hull@issinc.com

M. Workman (ed.), *Semantic Web,* DOI 10.1007/978-3-319-16658-2_6

asymmetric, reflexive, and disjoint properties and enhanced annotation capabilities, increase expressivity but require additional practitioner investment.

The purpose of this chapter is to characterize some of the significant gaps between NL and ontology semantics, and to articulate some of the challenges that these gaps present to the representation of knowledge extracted from text sources. It should be noted that we are not talking about how well we can or cannot do "text analysis," rather how well we can represent NL meaning as RDF or OWL knowledge representations.

The motivation for clearly defining these gaps is based on hard lessons learned in delivering semantic solutions to customers. These semantic gaps often represent major obstacles to meeting customer and user expectations, because the gaps are poorly understood, poorly communicated, or improperly addressed. If we can clarify these semantic challenges, we have a greater probability of agreeing on project requirements, functional expectations, and the next generation of research required to fill the gaps. A few customer-related problems are summarized here to provide a context for the more technical discussion below.

Challenge 1: Unrealistic Expectations
Customers, in our experience, do not understand how much "meaning" can be extracted from text sources using standard, ontology-based text processing, and ontology representation techniques. This can lead to disappointment, confusion, and, potentially, project failure.

Challenge 2: Confusing Terminology
The meaning of the term "semantics" is very different in NL (linguistics) and in ontologies (knowledge representation). Our customers—as well as the modelers themselves—have a difficult time distinguishing between the two definitions, a confusion that also leads to unrealistic expectations. We recommend specializing the term "semantics" into "NL (or linguistic) semantics" and "ontology semantics." The arguments for this differentiation are the subject of this chapter. As semantic technology professionals, we should be careful about defining these terms more meaningfully for customers, users, and ourselves.

Challenge 3: Semantic Modeling, Mapping, and Knowledge Extraction Shortfalls
The semantic expressiveness of ontologies simply is not sufficient to represent the semantic complexity of NL, at least not without building significant "representational scaffolding" to support it, leading to severe language-to-ontology mapping and modeling challenges. These challenges lead, in turn, to problems in extracting knowledge from text sources and representing it as ontology constructs.

To borrow an analogy from the film industry, editing NL semantics enough to fit into standard ontology structures requires us to leave a significant amount of valuable knowledge on the editing room floor. Understanding the semantic tradeoffs that must be made is critical for customers, information architects, and users, because meaning will be lost in the process of transforming NL semantics into ontology semantics, meaning that is often important to stakeholders. The remainder of this chapter outlines some of the major differences between "NL semantics" and

"ontology semantics." A good starting point for this comparative analysis is defining the different levels of semantic representation in an NL.

6.2 Levels of NL Semantics

In order to compare NL semantics to ontology semantics, it is important to define the primary linguistic structures that carry meaning. More specifically, we define NL semantics as the meaning expressed within NL at the morphological, lexical, syntactic, and discourse levels. We begin with a simple definition of "NL semantics," although a full definition, explanation, and defense of this definition is not possible within the scope of this overview. Following is a brief summary of the first three levels at which meaning is expressed in NL:

- Morphological level: morphology refers to the internal structure of words in an NL, consisting of their component "morphemes," the smallest meaning-bearing units in language. For example, the word "vehicles" consists of two morphemes: the root morpheme "vehicle" and the plural suffix "s."
- Lexical level: a language's vocabulary, including words and, perhaps, fixed expressions. Words have one or more morpheme, such as the two-morpheme word "vehicles."
- Syntactic level: rules for constructing meaningful, well-formed phrases, clauses, and sentences in a language, including permissible word order, such as "large air transport vehicles."

Meaning is expressed at all of these (and other) linguistic levels, sometimes conjointly, and in ways that are often quite difficult for ontologies to represent, at least in an ontology's standard form. Semantic translation problems at each level are summarized below.

6.3 Morphological Level

Definitions and Backgrounds

The two major classes of NL morphology are inflectional and derivational morphology. Inflectional morphemes change a word's grammatical category without changing its grammatical class. Examples from English include noun plurals (truck > trucks), verb tense (work > work*ed*), and verb aspect (go > go*ing*). In each case, if the uninflected form is a noun (e.g., cat), the inflected form remains one as well (e.g., cats); if it is a verb, it remains a verb in the inflected form, and so on.

By contrast, derivational morphemes change the grammatical class of a root word, say from a noun to a verb. Examples from English include changing from a verb to a noun (derive > deriva*tion*), from a noun to an adjective (derivation

> derivation*al*), or from an adjective to an adverb (derivational > derivational*ly*). English has a highly productive morphology, which allows speakers to easily create new words and meanings simply by using morphological rules of composition. The meaning represented in NL morphology poses many challenges in translation to ontology representations, discussed next.

Ontology Challenges: Morphology

The reason that morphological semantics causes semantic challenges for ontologies is that *ontologies do not have morphologies*. The myriad meanings represented via morphology in NL, therefore, are quite difficult to represent using standard ontology constructs. For example, a class (such as vehicle) is neither singular nor plural in an ontology, and a relation (such as drives) cannot be inflected for tense or aspect, as in *drove* or *driving*. The result is that these morphological meanings in NL (i.e., number, tense, aspect, etc.) are not easily translated into ontology constructs, unless extra work is done to extend standard ontology constructs to do so. A relation, such as: *manufactures* in the RDF triple::*person:manufactures:equipment*, is neither present, past nor future tense. Nor does it represent a completed, ongoing, or habitual activity. There is simply no natural way to "inflect" the relation for tense and aspect, as we do for NL verbs.

The point is that representing the meaning inherent in NL morphology is *unnatural* for an ontology at best, and therefore it is not normally represented at all. The inability to represent an NL's many inflected meanings in an ontology results in a many-to-one semantic mapping challenge in translating from an NL to an ontology, as shown in Fig. 6.1.

One might argue that the sentence patterns in Fig. 6.1 differ also by auxiliary verbs, such as *will, have*, and *be*. Two points counter this objection: (1) ontologies do not naturally handle auxiliary verbs, either, and (2) many languages express these meanings via inflectional morphemes, even if English does not. In either case, the point is the same: Multiple specific NL meanings get compressed into one generic canonical form in the ontology.

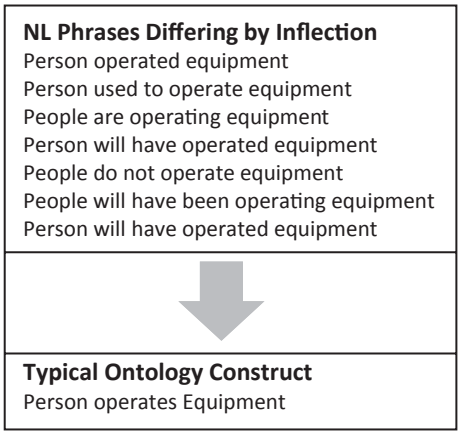

Fig. 6.1 Many inflectionally related NL words and meanings get compressed into one ontology construct and meaning, eliminating variations in meaning expressed by NL inflections, a process we call "morphological conflation." *NL* natural language

The point of this section is that significant meaning is expressed in NL quite naturally through its morphology, or closely related auxiliary verbs. Through morphemes, we can describe an activity as past, ongoing, habitually repeated, or a future improbability. We can identify subjects or objects that are individuals, couples, or groups. We can articulate whether an action is completed, ongoing, or not started at the time referenced. By adding modal verbs (such as *must* or *may*) and adverbs, we can discuss possibility, probability, certainty, or impossibility. And we can very flexibly combine these meanings (via morpho-syntactic rules) to express complex meanings like "The driver *most probably will have been operating* equipment by tomorrow morning." Without significant work, each of these NL grammatical patterns and meanings, when extracted from text, will be reduced to one canonical form, or assertion, in the knowledge base, such as *::person:operates:equipment*—leaving representation of the other morphological meanings (or senses) on the cutting room floor. For some applications, this may be quite acceptable, such as simply finding text sources that seem to be "about something" in general. In others, such as a global disease spread application, it would be advantageous to differentiate projected future state from reported past state, or the conditions under which the disease spread may be expected to increase or decrease.

A more extensive set of "lost meaning" examples is presented in Fig. 6.2, which illustrates the semantic downsizing required to map the semantics of NL morphol-

NL Morphology	Examples	Meaning	Ontology Construct
Inflections			
Tense	was located	past	
	is located	present	
	will be located	future	
Tense & Aspect	was being located	past progressive	locate (relation)
	is being located	present progressive	
	will be being located	future progressive	
	will have been being located	future past progressive	
	(and so forth)		
Number	location	singular	location (class)
	locations	plural	
Gender	he	masculine PN	
	she	feminine PN	
	they	non-gendered plural PN	*x no pronouns;*
Person	I	first person	substitute instance
	you	second person	name
	he	third person	
Possession	his	possessive PN	
	person's	possessive noun	Anton posseses X
	Anton's	possessive personal N	(assertion/relation)
Derivations			
	locate	verb	
	location	noun	location (class)
	locational	adjective	locate (relation)
	locationally	adverb	

Fig. 6.2 Many NL morphological variations are conflated to a single structure and meaning in an ontology. *NL* natural language, *SVO* subject–verb–object

Natural Language	Conflation Type	Ontology	Description
Morphological variants: ➢ Send ➢ Sent ➢ Sends ➢ Sending ➢ Manufacturer ➢ Manufacturers	Morphological conflation	Single canonical form ➢ Sends ➢ Manufacturer	Many inflectionally related forms in a NL are conflated to one morphological form in an ontology.
Phase of words: Steel manufacturer sends steel products to construction customers.	Lexical conflation	Single string literal forms: Class: steel-manufacturer Property: sends-steel-products-to Class: construction-customer	Multiple NL words are conflated to one ontology string for each literal (S, V and O)
Complex sentence syntax: Steel manufacturer sends steel products to construction customers.	Syntactic conflation	Simple SVO syntax: S: Steel-manufacturer V: sends-steel-products-to O:construction-customers	Complex NL phrase structures are conflated to simple SVO ontology statements (triples).

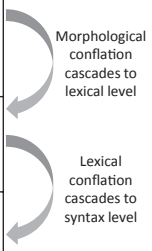

Fig. 6.3 Cascading conflation is produced by the effects of morphological conflation on the lexicon, and the effects of morphological and lexical conflation on the syntax of an ontology

ogy into simpler ontology constructs. Although this example set is clearly incomplete, it illustrates the many meanings in NL morphology that must be compressed into the simple semantics of an ontology, some of which are briefly discussed below. In addition, this corpus illustrates the many different word forms that are downsized to one canonical form in the translation from NL to an ontology.

We call this downsizing process "conflation," which has a dictionary definition of "a merging of diverse, distinct, or separate elements into a unified whole" (Dictionary.com 2014). In these examples, the various senses represented by morphological variations are conflated to one "ontological sense," and the word structures expressed by these morphological (and auxiliary verb) variants are conflated into one "ontological form." In other words, an ontology modeler will usually represent "go, goes, gone, going, went, will be going, will have gone, will have been going" as a single form with a single sense, such as "goes." To differentiate between sense and form conflation in discussing their impacts on ontology modeling, we use the phrases "semantic conflation" and "structure conflation." However, both are the result of the same root cause: The fact that an ontology has no morphology.

As we shall see, conflation at the morphological level has a cascading effect into the lexical and syntactic levels, since these linguistic levels are related in sophisticated ways. The end result is "cascading conflation" into the lexical and syntactic levels of ontology representation (Fig. 6.3), which we explore in the following sections. But, first, we discuss a few more semantic modeling challenges at the morphological level.

Related Word Challenge

Words related by morphological rules in NL must be represented as separate, unrelated constructs in an ontology, if they are represented at all. There is no principled way to "derive" a new concept by adding a meaningful morpheme, such as by adding an [-er] suffix to the verb *manufacture* to derive the noun *manufacturer*. If these two concepts are represented in ontology, they are represented as independent,

unrelated terms, in the morphological sense. The ontology simply does not know that these terms share a common root meaning.

At best, their relationship could be expressed in that the class manufacturer might be modeled as the domain for the relation *manufactures*, as *in::manufactures rdfs:domain:manufacturer*, but this assertion states that *manufacturer* can be the "subject" of the relation *manufactures*, not that these two words share a common meaning. The shared "sense" between the two is simply lost in the ontology. In order to simplify presentation in the remainder of this chapter, we use a pseudo-code shorthand notation for expressing RDF triples, as in: *manufacturer manufactures product*, rather than using the standard RDF syntax::*manufactures rdfs:domain:manufacturer* and:*manufactures rdfs:range:product*.

Tense and Aspect Challenges

While we may make existential statements in an ontology, such as: *person operates equipment*, we are severely limited in representing whether this assertion is past, present, or future tense, as in "Rich operated the forklift" (past) or "Rich is operating the forklift" (present progressive). The meaning associated with such inflections is hard to represent in typical ontologies. In order to do so, temporal semantics must be added to the ontology (Hobbs and Pan 2006), along with the logic needed for inferencing about time, based on time–date property values. Although we can time-stamp a fact in RDF, such as: *event occurs-on date*, we must apply axioms to determine whether that event is a historical or future fact relative to any other assertion in the knowledge base. Tense is not carried in the relation itself, as it is in the NL (English) verb.

Ontology Modeling Challenges

Modelers must select class and property names from among the many possible inflected forms in naming ontology literal constructs. The following two triples are typical examples:

- Ontology 1: *Person operates equipment*
- Ontology 2: *People operate equipment*

In both cases, modelers intend to represent *person* and *equipment* classes and an *operate* relation that semantically connects them. Therefore, from a conceptual viewpoint, these two statements are semantically equivalent, a fact that is evident only to morphologically competent humans. In an ontology, assertions must be made—or algorithms must be developed—to identify this semantic equivalence; the semantic mapping is unnatural to the native ontology representation. For example, one might use the *equivalentClass* relation to indicate person and people are "synonyms," as in: *Person equivalentClass People*. However, this in no way represents the meaning

inherent to the NL plural morpheme; in an NL, these are not equivalent words, as they become in the ontology, another example of morphological conflation. Ontologies simply do not have a natural way to represent morphological meaning variations of this kind.

Morphology Challenge Summary

Morphological conflation results in a significant meaning loss between an NL and ontology. In addition, it has a cascading effect at the lexical and syntactic levels, which we shall discuss momentarily. This meaning loss has significant ramifications for customer expectations. Whether in financial market analysis or military intelligence analysis, there is a significant difference between a fact in the past, present, or future tense. Knowing a stock merger will happen, or that an attack may occur, are very different than stating that they already happened, or that they probably will not occur. Yet, to represent these important meaning differences in an ontology requires a significant amount of additional "representational scaffolding," such as temporal extensions to ontology standards (Hobbs and Pan 2006), assertion of additional facts, or the addition of logic to handle comparative inferences among assertions, and so forth.

We discuss the approaches we are taking to extend an ontology to model morphological semantics in Chap. 7.

6.4 Lexical Level

Background and Definitions

There are significant semantic gaps between an NL and ontology at the lexical (or word) level as well. The morphological gaps between an NL and ontology play a significant role in that difference, as mentioned earlier, since people use morphological rules to create related words. These are therefore interrelated conflation challenges. An NL's lexicon, or vocabulary, can be roughly categorized into two primary types of words: content words and function words. Content words are those with lexical meaning (such as *car, cake,* or *careful*) while function words are those that relate one grammatical structure or meaning to another in some way (such as *and, then, will, therefore, he* or *of*). Although function words have no content, they do provide contextual meaning, expressing notions like conjunction, exception, direction, definiteness, or previous reference.

The set of function words is a "closed set," in that there are a set number of pronouns, prepositions, or articles in an NL. Whereas content words contain most of the domain meaning in an NL at the lexical level, function words are critical to how people understand the complex, meaningful relationships among content words,

phrases, and clauses. Function words allow NL speakers to create increasingly complex, but meaningful, phrase and sentence structures, much as morphology does within words.

The problem is that an ontology has no function words, just as it has no morphology. It has no conjunctions to join two words, phrases, or clauses together to form conjoined subjects and objects or compound sentences, at least at the instance level of ontology representation. In addition, an ontology has no pronouns to support anaphora, or previous reference. It has no prepositions to turn nouns into modifiers of location, direction, or time. And it has no helping verbs to vary the tense, aspect, or modality of a verb phrase, as previously discussed. In summary, the lack of function words in an ontology makes it unnatural to represent these kinds of connective, directional, and referential semantics natural to languages, as exemplified by the italicized words in the following sentence: Tom *and* Alice drove *to* town *and then they* walked *to the* mall *to* buy clothes *and* see *a* movie *before* returning home *to* eat dinner, *after which they* went *for a* walk.

In this one example sentence, there are over ten kinds of "connective semantics" represented by function words like *and, to, then, they, which,* and *before,* as well as the phrases that these function words introduce, any one of which is difficult for RDF/OWL representations, especially in the A-Box (e.g., at the instance level).

The lexicon of an ontology is quite different from that of an NL in another way. Whereas NL words are meaningful to people, ontology words are not really meaningful to a computer, not in any linguistic sense (Manola et al. 2014). Ontology words consist of the strings that make up their class, property, and instance names. The fact that these strings "look" like NL words or phrases to people results from the fact that modelers usually choose to use strings based on NL words and phrases to help other people understand what sense they mean to express with a specific string. Within the ontology, however, a class named "equipment" could just as well be represented as the string "X" as far as the computer, RDF, or OWL are concerned, as long as it is unique within its own namespace (Manola et al. 2014). This leads to widely varying string names within different, but related, ontologies. Modelers are free to name a class or property whatever they like, as long as it is unique.

A third difference between a NL and ontology lexicon is that an ontology's words are not related by morphological rule, as NL words are in word sets such as *begin*, *beginner*, and *beginning*. Here we see how the lack of morphology has a significant representational impact on an ontology's lexicon. Each ontology "word" is semantically independent from all other words. This stands to reason: since an ontology has no morphology, there is no natural way to "create" new, related words via morphological rule, nor to represent the meaningful relationships among these inflected and derived word forms in the lexicon, as is done in an NL dictionary. Morphologically related words with a common core meaning—like create, creator, creative, and creation—have no more meaning in common in an ontology's dictionary than do words like disease, justice, drink, and sunrise, at least morphologically speaking. While one could imagine expressing morphologically related ontology constructs with RDF (pseudo-code) statements, such as:

- Creation noun-derived-from create
- Created past-tense-of create
- Creator creates creation

not only are these awkward and unnatural assertions but they also create other knowledge representation problems. For example, the first of these assertions uses the property *create* as the object of a statement, treating it as an individual and therefore pushing the overall OWL ontology into the OWL Full language, hindering description logics (DL) reasoning (Bechhofer et al. 2004). In addition, since there are no morphological rules of word composition, every morphological variation among words would need to be expressed as an individual fact, such as:

- Wanted past-tense-of want
- Created past-tense-of create
- Creating progressive-aspect-of create

This leads to an exploding knowledge base, if nothing else. As a final remark, it is hard to imagine how *creating* would then be used within the ontology, since the modeler wants to treat it as a present progressive verb form (a relation in the ontology), yet it has just been rendered a class within the ontology, which functions more as a noun form. This is a difficult challenge to address within the natural constructs of an ontology.

A final difference between NL and ontology lexicons is that an ontology's "words" often consist of what would be an entire phrase in an NL, such as the class name *major-launch-processing-operation* or the object property (relation) name *is-the-subject-of*. This string concatenation results from the fact that an ontology limits each class or property name to a single string, or literal, rather than a sequence of words making up a phrase. This "word concatenation" phenomenon represents the next level of cascading conflation—*lexical conflation*—which is discussed further in the challenges below.

Synonym Challenge

The three example assertions just presented point out another related semantic challenge for ontologies at the lexical level: Synonym identification and representation. Synonyms are two words with (roughly) the same meaning, at least within a specific communication context. Synonymy, natural to languages, is quite unnatural in an ontology. In this example, NL speakers may interpret *manufacturer, company,* and *organization* as roughly synonymous nouns and *develops, manufactures,* and *produces* as roughly synonymous verbs, given the context. Ontologies, however, do not naturally account for synonymy without an explicit assertion, such as (in pseudo-code): *Manufacturer same-as company*.

This representation is at best an inelegant and inefficient way to represent synonymy, but at least it is available within the representation standard. However, the

challenge is greater for ontology class/property names that are based on concatenating words from NL phrases. Examples abound, such as the class name *Action-Temporal-Association* and the relation name *is-acted-upon-as-specified-by*, both from the US Department of Defense's JC3IEDM standard data model (Morris 2012). These are examples of lexical conflation, in which multiple NL word forms with multiple related senses are conflated to a single ontology form and sense. Since ontology constructs like these are not composed of individual, meaningful words, it is very difficult to map their meaning to NL paraphrases in text (such as the paraphrase: *Event relationship to time*). To do so would require a word-by-word, sense-by-sense comparison at the lexical level. To put it another way, ontology names are not "lexically based" but rather "string based." This example class name is not a combination of the three NL words *action, temporal,* and *association,* each with its own entry and meaning in the lexicon, and each belonging to a specific grammatical category (e.g., noun, adjective). Rather, the NL words were simply used by human modelers to form a string name, or literal, that looks as if it is formed from these words.

Lexical conflation, again, is caused by an ontology's lack of grammar, which is a set of rules that humans use to creatively construct phrases and sentences from individual words. Whereas in an NL, any number of words can work together as the subject of a sentence, for example, an ontology is limited to "one string," equivalent to one word in the ontology's lexicon. This limitation makes it very difficult to use conflated ontology concept names as a basis for finding NL paraphrases in text. The "one string–one meaning" restriction does not map well to "multiple words–multiple meanings" of an NL, at least without additional data structures and algorithms to support the analysis. Clearly, lexical conflation is a challenge that spans the lexical–syntactic boundary, and is discussed further in the section "Ontology Challenges: Syntax."

The implication of the synonym and paraphrase "gaps" between an NL and an ontology is that it is very difficult, using ontology semantics only, to identify, capture, and address synonymy and paraphrase in NL text. These gaps present major problems in extracting knowledge from text and transforming it into an ontology-based representation. Some of the knowledge extraction and representation problems that arise from this gap include:

- Difficulty in identifying and representing different words (from NL sources) with the same meanings (synonymy).
- Difficulty in identifying and representing different phrases with the same meanings (paraphrase) as lexically conflated ontology names.
- Difficulty in preventing redundant information extracted from NL text from being asserted to the knowledge base, since the redundancy is expressed by different words that mean the same thing in text.
- Difficulty in semantically integrating, or fusing, semantically related textual information, where the NL semantics cannot be naturally represented in the ontology.

Function Word Challenge in the Lexicon

The absence of function words in an ontology, such as articles (an, the), pronouns (he, she), or prepositions (of, from), makes it difficult to connect classes, properties, and triples in meaningful ways, particularly at the instance, or A-Box, level. It is difficult to express conjoined events and objects (and), conditionals (if this, then that), directionality (to the store), previous reference (the, he), or concurrency (while, during, when).

The fact that ontologies lack function words adds to the cascading conflation problem, contributing to some odd naming conventions in ontology modeling. Single NL phrases introduced by function words, such as prepositional phrases (PP), must often be split into two parts, with half of the phrase used in one construct's string name (e.g., a relation name) and half used in another's string name (e.g., a class name). Consider the following examples taken from the JC3IEDM standard (Morris 2012):

- Action is-the-subject-of Action-Functional-Association
- Action is-acted-upon-as-specified-by Organization-Action-Association
- Action is-geometrically-defined-through Action-Location

In the first example, the NL PP version of the relation (of action functional association) serves an adjectival role, modifying the predicate nominative subject. However, ontologies have no modifiers, such as adjectives, adverbs, or phrases that serve those roles. Within the ontology, the preposition "of" is therefore modeled as part of the relation name (*is-the-subject-of*), whereas the object of the preposition in the NL phrase is used to model the class name (*action-functional-association*). The NL PP has been split into two parts in the ontology, with one part used in the conflated relation name, and the other part used in the conflated class name.

Clearly, this "NL phrase splitting" is only part of the cascading conflation effect, since the predicate nominative (e.g., "the subject") is also conflated into the relation string name. This approach to naming ontology constructs results from the constraints placed on modelers at the ontology's syntactic level, a problem that is discussed further in the "Syntax-Level Challenges" section.

Lexical Challenge Summary

The lexicon of an NL and that of an ontology are significantly different in both form and meaning. One could claim that ontologies do not really express lexical meaning at all, at least not in any NL sense of the term. Even if we accept that they do, the depth and breadth of lexical meaning are highly restricted relative to NL for the reasons discussed. We present potential approaches to dealing with semantic gaps at the lexical level in Chap. 7.

6.5 Syntax-Level

Background and Definitions

An NL typically has a sophisticated syntax used to sequence words, phrases, and clauses to produce meaningful sentences. Syntax is governed by grammatical rules that define well-formed phrases and sentences for the language, allowing speakers to build words from morphemes; phrases and clauses from words; and sentences from phrases and clauses. Multiple words can be combined into a single grammatical construct, which can serve as a subject, predicate, or direct object of a sentence.

A subject in an NL is frequently a phrase with many words combined dynamically to express complex meaning; a subject in an ontology is always "one string." Humans understand that a combination of words in NL can be used together as a single subject or object (single referent), but that each of the constituent words has its own meaning in the lexicon as well, used within the phrase to specialize the meaning of the referent. That is, humans can parse these phrases into their meaningful words, and the words into their meaningful morphemes, based on shared rules of grammar. By so doing, they are also able to compare the meaning within these phrases to components of other phrases for similarities and differences. In that manner, people can easily determine that "minor launch processing operation" is similar to "major launch processing operation," perhaps different only in complexity or duration, based on the semantic difference between the words "major" and "minor."

Ontologies are quite different in this respect, as we have discussed. First, they have a very limited syntax. Second, they have no grammatical rules with which to construct phrases and clauses from individual words, just as they have no morphological rules for constructing new words from morphemes. These limitations have a significant impact on ontology modeling, as well as on mapping NL phrases and sentences to ontology constructs. Each is discussed in turn.

An ontology's syntax is typically limited to simple subject–verb–object (SVO) sentences (devoid of morphology), as in: *Mechanic repairs equipment*. These are called "triples" because they are assertions of exactly three ontology "words," each word a string, that are used to represent domain knowledge. Roughly, these are equivalent to simple active and passive voice sentences in an NL, with some significant limitations, however. We return in a moment to the challenges presented by this limited syntax.

Equally limiting is the fact that ontologies have no grammatical rules for dynamically creating new "multi-word" constructs, such as new class and property phrases, as discussed above. Therefore, ontologies cannot build phrases out of a sequence of words, and their associated meanings. For example, NL users can combine the words *lacrosse, sports,* and *equipment* into *lacrosse sports equipment,* and other speakers will immediately understand the phrase as composed of the constituent words and their individual meanings. In an ontology, *lacrosse-sports-equipment* is not composed of three words with three constituent meanings; it is one conflated

string with one conflated meaning. Another class, say *sports,* is quite unrelated to it. The constituent word meanings are lost on the ontology, which has no grammatical rules to parse this string into constituent word structures and senses. Each string, to an ontology, is a "single canonical form" rather than a multi-word phrase, where each constituent word has a contributory meaning.

These two syntactic limitations result in syntactic conflation, in which myriad NL phrase and sentence structures must be transformed into simple SVO structures, with each S, V, and O represented by one (and only one) "ontology word." This is a serious restriction if our objective is to represent NL meaning within ontology semantic structures. A few of the impacts that these syntactic/semantic constraints have on ontology modeling include the following:

- Complex NL syntax must be transformed into a set of simple SVO sentences, or triples.
- Each construct in an SVO triple consists of one and only one named element, rather than an unlimited sequence of meaningful words combined into a phrase.
- Each construct in a triple is devoid of morphology, meaning that the S, V, and O do not vary in meaning, such as tense changes for verbs (relations) or singular/plural for subject and objects (classes).
- The lack of function words likewise makes it difficult to meaningfully connect one SVO triple to another to make compound sentences or to construct a discourse sequence, as in: Kent and Mark ate food. *Then* they played golf *before* they went to the movies.

In linguistic terms, this is like limiting NL sentences to noun–verb–noun structures, where the nouns and verbs can be at most one word in length and have no internal morphology to vary meaning. These combined limitations severely restrict the meaning that can be expressed naturally with traditional ontology syntax. The net result of cascading conflation at all of these linguistic levels is that modelers must fit some very large square pegs (NL semantics) into some very small round holes (ontology semantics), leading to significant meaning loss in the translation, as well as some rather bizarre modeling constructs.

Ontology Challenges: Syntax

Significant semantic content expressed by NL phrases and sentences will be lost in transforming them into an ontology's simple SVO syntactic structure, with morphological and lexical limitations adding significantly to the cascading conflation effects, as previously discussed. However, modelers use ontology conflations because they are attempting to meet two important but incompatible requirements: (1) to make an ontology name a single string representing a single class/property concept (a data structure modeling requirement) and (2) to make the string look like a normal, syntactically correct sequence of NL words that expresses the true mean-

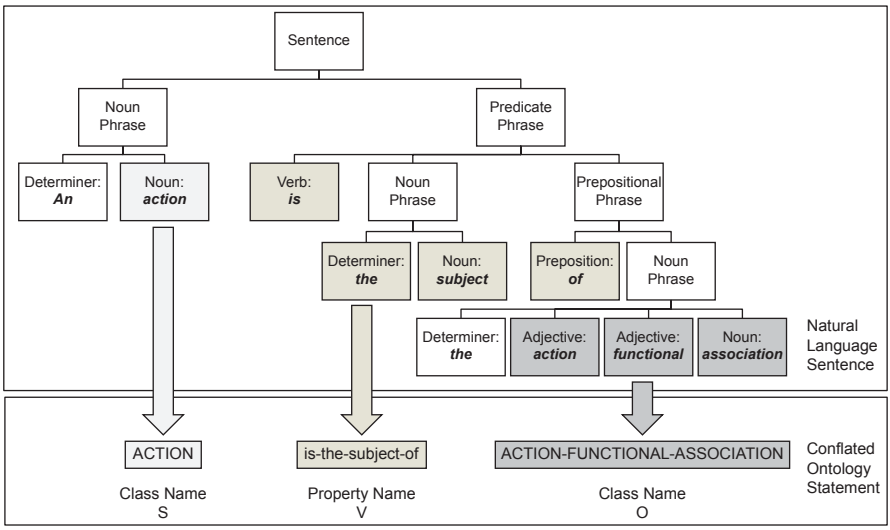

Fig. 6.4 Complex NL phrase and sentence structures. (Note: Complex NL phrase and sentence structures are conflated into three strings in an ontology, representing a simple SVO structure, with each element consisting of a single "ontology word.") *NL* natural language

ing of the string, so that people can understand the concept (a human understanding requirement). The resulting syntactic conflation, however, creates a serious confusion for those who do not understand that the ontology string names are not actually composed of individual NL words, as they appear to be.

Figure 6.4 illustrates syntactic conflation across all elements of a SVO triple, based on the first JC3IEDM example presented above, as it relates to the syntactic structure of the equivalent NL sentence. In effect, the NL syntax tree is flattened into three ontology strings, representing its SVO (or Class Relation Class) structure. The original NL sentence loses its internal constituent phrase structure. This results in the "phrase splitting" phenomenon discussed above, in which prepositions and predicate nominatives, for example, are conflated into the relation name, while the object of the preposition is conflated into the class name. The entire sentential structure has been conflated to three ontology "words" within this simple SVO triple.

6.6 Summary and Value Proposition

In summary, it is important to understand the significant differences between NL semantics and ontology semantics in order to level-set expectations for customers, users, modelers, and other stakeholders. Standard ontology constructs are too restrictive in structure and semantics to naturally represent the range of meaning that can

be expressed in a NL, leading to cascading conflation problems in translating NL morphological, lexical, and syntactic meanings into ontology semantics. We have illustrated some of the types of "semantic gaps" that exist between an NL and ontology, and summarized some of the typical conflations resulting from those gaps. Given the complexity of mapping NL semantics into ontology semantics, readers may ask a legitimate question: Why extract ontology-based knowledge from text at all? Where is the value proposition?

A thorough answer to this question is beyond the scope of this short overview, and is not the main purpose of this chapter, but the benefits are significant and worth a summary comment. Here are only a few of the ways in which NL-to-ontology semantic translation can be scoped and extended to be highly useful:

- Extract only highly relevant essential elements of information (EEIs) from text. This might include specific event types, individuals, or locations, for example. This can be accomplished effectively for focused analysis needs by extending ontologies' domain-specific vocabularies and grammars, often available as open-source tools (Cunningham 2014).
- Relate extracted EEI's to each other based on the ontology. These connection graphs provide a way to semantically link, or fuse, EEIs extracted from heterogeneous sources based on shared concepts and relations.
- Extract and classify data based on a taxonomy, which provides more generalized search over text and data.
- Build rules for deductive reasoning over RDF knowledge bases, providing a way to infer new facts based on known facts.
- Use ontologies to publish the meaning of information for discovery by Semantic Web services.

In our follow-up chapter, we address some of the challenges that have been discussed in this chapter. In each case, our research objective is to bridge the gap between NL and ontology semantics, creating a more "natural" ontology representation of NL semantics.

References

Bechhofer, S., et al. (2004). OWL web ontology language reference, W3C recommendation. http://www.w3.org/TR/owl-ref/. Accessed 9 Nov 2014.

Bimson, K. D. (2009). Principles of interontology development, research supporting lingua franca requirements and design, tech. report, Modus Operandi.

Cunningham, H., et al. (2014). Developing language processing components with GATE version 8 (a user guide). University of Sheffield Natural Language Processing Group. https://gate.ac.uk/sale/tao/#sec:howto:plugins. Accessed 9 Nov 2014.

Dictionary.com. (2014). Dictionary.com. http://dictionary.reference.com/browse/conflation. Accessed 9 Nov 2014.

Hobbs, J. R., & Pan, F. (2006). Time ontology in OWL, W3C recommendation. http://www.w3.org/TR/owl-time/. Accessed 9 Nov 2014.

Manola, F., Miller, E., & McBride, B. (2014). RDF 1.1 primer, W3C working group note. http://www.w3.org/TR/2014/NOTE-rdf11-primer-20140624/.
Morris, M. (2012). Multilateral interoperability programme, the joint C3 information exchange data model (JC3IEDM). Greding, Germany.OWL 2 web ontology language, W3C recommendation. http://www.w3.org/TR/owl2-overview/. Accessed 9 Nov 2014.
RDF—Semantic Web Standards. (2004). RDF—semantic web standards W3C recommendation. http://www.w3.org/RDF/. Accessed 9 Nov 2014.