Language and its Commonsense: Where Formal Semantics Went Wrong, and Where it Can (and Should) Go

Walid S. SABA*

Ontologik.ai

Journal of Knowledge Structures & Systems

November 2020

Vol.: 1 Issue: 1 Pages: 40-62

Abstract

The purpose of this paper is twofold: (i) we will argue that formal semantics might have faltered due to its failure in distinguishing between two fundamentally very different types of concepts, namely *ontological concepts*, that should be types in a strongly-typed ontology, and *logical concepts*, that are predicates corresponding to properties of, and relations between, objects of various ontological types; and (ii) we show that accounting for these differences amounts to a new formal semantics; one that integrates lexical and compositional semantics in one coherent framework and one where formal semantics is embedded with a strongly typed ontology; an ontology that reflects our commonsense knowledge of the world and the way we talk about it in ordinary language. We will show how in such a framework a number of challenges in the semantics of natural language are adequately and systematically treated.

Key words: Commonsense knowledge; Background knowledge; Strongly-typed ontology; Language understanding; Formal semantics

1 Introduction

In the concluding remarks of *Ontological Promiscuity*, Hobbs (1985) made what we believe to be a very insightful observation: given that semantics is an attempt at specifying the relation between language and the world, if "one can assume a theory of the world that is isomorphic to the way we talk about it ... then semantics becomes nearly trivial". But how exactly can we rectify our logical formalisms so that

*✉ walid@ontologik.ai

semantics, an endeavor that has occupied the most penetrating minds for over two centuries, can become (nearly) trivial, and what exactly does it mean 'to assume a theory of the world' in our semantics?

In this paper, we hope to provide answers for both questions. First, we believe that a commonsense theory of the world can (and should) be embedded in our semantic formalisms resulting in a formal semantics grounded in commonsense metaphysics. Moreover, we believe that the first step to accomplishing this vision starts by rectifying what we think was a crucial oversight in formal semantics, namely the failure to distinguish between two fundamentally different types of concepts: (i) *ontological* concepts, that are **types** in a strongly-typed ontology; and (ii) *logical* concepts, that are **predicates** corresponding to properties of and relations between objects of various ontological types. By embedding ontological types in our predicates, type unification and other type operations can then be used to 'uncover' missing information, i.e. information that is never explicitly stated in everyday discourse, but is often implicitly assumed as shared background knowledge.

In the next Section, we first discuss what we call the 'missing text phenomenon' (MTP) that–we argue–is behind most challenges in the semantics of natural language. We will then discuss the difference between ontological and logical concepts, and how acknowledging this difference effectively translates into embedding a commonsense theory (knowledge) of the world into our logical formalism. Subsequently, it will be demonstrated how differentiating between logical and ontological concepts can indeed make semantics become 'nearly' trivial by first suggesting how type unification over predicates embedded with various ontological types can help us 'uncover' all the missing text, i.e. text that is never explicitly stated but is usually implicitly assumed as shared background knowledge. Subsequently, we will show how several challenges in the semantics of natural language (e.g., lexical disambiguation, metonymy, copredication, etc.) can be adequately and *uniformly* treated by accounting for the missing text. Before we proceed, however, we define here what we mean by two important phrases that are mentioned throughout in this paper:

Commonsense knowledge: That knowledge that is available for a 4-year old, and not the sort of domain specific knowledge that might be needed in problem solving in specific domains; e.g., 'trees don't walk', 'it is not sensible to say articulate table'; in 'The White House issued a warning to North Korea' The White House is a reference to the president and his administration, not the building, etc.

Background knowledge: That knowledge that is not explicitly stated in everyday discourse but is implicitly assumed to be shared by speaker and listener. For example, a bartender hearing a waiter in a bar saying 'the corner table wants another beer' knows that it is '[the person sitting at] the corner table that wants another beer' and specifically because of the shared background knowledge that people and not tables drink beer; that the most salient relationship between a person and a table is the 'sitting-at' relation, etc.

2 The Missing Text Phenomenon (MTP)

Linguistic communication happens, crudely, as follows: a thought is encoded by a speaker into a linguistic object (in some language), and the listener then decodes the linguistic object into (hopefully) the thought the speaker intended to convey. In this complex process, there are several alternatives how this can happen: (i) on one extreme, the speaker can compress (and minimize) the amount of information sent in the encoding of the thought and hope that the listener will do the extra work in the decoding (uncompressing) process to get at the message being conveyed; or (ii) the speaker will do the hard work and send all the information needed to convey the thought, which would leave the listener with little to do (a detailed description of this process can be found in Kirby et al., 2015).

The natural evolution of this process, it seems, has resulted in the right balance where the total work of both speaker and listener is optimized. That optimization resulted in the speaker encoding the minimum possible information that is needed, while leaving out everything else that can be safely assumed to be information that is available for the listener–information that we usually call *common* background knowledge. While this genius optimization that has developed in about 200,000 years of evolution works quite well for humans, it is also precisely why natural language understanding (NLU) is difficult: machines do not know what we leave out and implicitly assume as shared background knowledge. To illustrate how this 'missing text phenomenon' (MTP) manifests itself in well-known challenges in NLU, consider the examples below where the missing text is highlighted in **bold**:

LEXICAL AMBIGUITY
I like to play bridge
⇒ *I like to play* [**the game**] *bridge*

HIDDEN RELATIONS
Mary enjoyed the (*sandwich* | *movie*)
⇒ *Mary enjoyed* ([**eating**] | [**watching**]) *the* (*sandwich* | *movie*)

QUANTIFIER SCOPE AMBIGUITY
BBC has a reporter in every country
⇒ *BBC has a* [**different**] *reporter in every country*

PREPOSITIONAL PHRASE ATTACHMENTS
Jon had pizza with (*pineapple* | *his kids*)
⇒ *Jon had pizza* ([**topped**] | [**together**]) *with* (*pineapple* | *his kids*)

METONYMY
The corner table wants another beer
⇒ [**the person sitting at**] *the corner table wants another beer*

METAPHOR
Don't worry about Peter, he's a rock
⇒ *Don't worry about Peter, he's* [**solid like**] *a rock*

NOMINAL COMPOUNDS
Jon works in the (*computer* | *neighborhood*) *store*
⇒ *Jon works in the* (*computer* [**-selling**] | *neighborhood* [**-located**]) *store*

COPREDICATION
Jon threw the newspaper after he read its criticism of globalism
⇒ *Jon threw the* [**physical-object**] *newspaper after*
he read its [**informational-content**] *criticism of globalism*

What all the above indicates is that most challenges in NLU are due to the challenge in uncovering all the information that is not explicitly stated but is often assumed as shared (and common) background knowledge. As noted by Levesque (2011) in the context of the Winograd Schema Challenge, in order to have a full understanding of ordinary spoken language "you need to have background knowledge that is not expressed in the words of the sentence to be able to sort out what is going on ... And it is precisely bringing this background knowledge to bear that we informally call *thinking*" (emphasis in original).

In formal semantics, however, there has been quite a bit of work in recent years to develop type-theoretic semantics to deal with some instances of this phenomenon. For example, in a series of papers (see e.g. Lou, 2010; 2011; 2012) Lou introduces a type system based on Martin-Löf's type theory (Martin-Löf, 1984) where common nouns are considered to be types, and where it is shown how the machinery of type coercion can in such a system handle lexical disambiguation as well as accommodate for what is referred to as copredication (see Pustejovsky, 1995), which occurs when an object is predicated in different ways in the same context. For example, the context in (1) is one where 'book' is used in two senses in the same context: as the physical object (when being 'bought') and as an INFORMATIONAL CONTENT (when being 'read'):

(1) *John bought and read the latest book on deep learning*

While we are sympathetic to the general approach of Lou, we believe that copredication and lexical ambiguity are in fact part of the same and much simpler phenomenon, and thus we believe that copredication and type coercion introduce complex machinery unnecessarily, not to mention that type shifting/coercing will not always produce the desired results. The same observation can be made about the work of Asher and Pustejovsky (2012) where complex machinery that permits type shifting is used to access different aspects (senses) of a structured object (or a dot-type, that represents all possible senses) using lexical constraints available in the context. The problem we have with this approach is that the notion of a dot-type does not seem to be cognitively plausible, not to mention that, in theory, language allows us to pick many aspects of a given object that cannot *a priori* be defined as part of the lexical semantics, but must be dynamically (compositionally) figured out (we will see this in some detail below). Moreover, and more specific to the work of Lou, we argue that there is in fact a technical problem in assuming that the entire class of common nouns should constitute the types in the system. For example, while nouns such as 'man' and 'bank' can reasonably be treated as types, this will not do in situations where the common noun is a role noun (e.g., teacher), as illustrated in (2).

(2) *John is an excellent teacher*

In (2), it is not 'John' (the teacher) that is excellent, but John's teaching (ACTIVITY) instead.

Starting with (Asher, 2008; 2011) and more recently (Asher, 2015), Asher has also developed over a few years a type system that is aimed at incorporating some commonsense metaphysics through a typed system. While the same reservations we have regarding the approach taken by Lou (2012) more or less apply to the earlier work of Asher, the more recent Asher (2015), however, correctly highlights the technical problems in simply performing type shifts (or type coercion), and in particular in examples such as those in (3).

(3) *Julie bought a book. It was a mystery.*

If a straightforward type shifting is performed on the first sentence, so that the type constraints imposed by 'enjoy' (which expects a PHYSICAL-OBJECT) are satisfied, then the subsequent sentence cannot be correctly interpreted as we would have lost, so to speak, the INFORMATIONAL-CONTENT sense of book that is the obvious referent of 'it'. Asher concludes, and correctly so, in our opinion, that it is not type shifting of book that must occur, but that some process in predicate composition must occur. That process, which Asher (2015) calls 'transformation,' is essentially a functor (a function object, something similar to a lambda expression, in computing lingo), that 'picks-up' the desired object sense that can semantically link to the verb's argument. We are in agreement with the spirit of this approach but we have two reservations. For one thing, this 'transformation' operation is not very clear, especially in how it picks-up different kinds of objects, e.g., an eventuality (a 'reading') in (4) and 'informational content' in (5):

(4) *Julie enjoyed the book* ⇒ *Julie enjoyed* **reading** *the book*

(5) *Julie criticized the book* ⇒ *Julie criticized* **the content of** *the book*

Moreover, and if types are meant to be a set of general categories (such as PHYSICAL-OBJECT, INFORMATIONAL-CONTENT, ANIMATE, etc.), it is not clear how it can be determined that in (6) it was Barcelona's **residents** who voted for independence (6a), that it was Barcelona's **team** that lost to Real Madrid (6b), and that it was Barcelona's **governing body** that announced a curfew (6c).

$$(6) \qquad Barcelona\ was\ calm\ after\ it \begin{cases} \text{a. } voted\ for\ independence \\ \text{b. } lost\ to\ Real\ Madrid \\ \text{c. } announced\ a\ curfew \end{cases}$$

The reason this highly general (and somewhat *ad-hoc*) type system is potentially problematic is that all the [**missing terms**] in the examples in (6) are, at a high level, of a similar type, namely some 'group of people' and thus for our pragmatics to work it must operate at a much more granular level, as we will see below.

To summarize, the recent efforts to incorporate type-theory in compositional semantics in an effort to integrate lexical semantics with compositional semantics is a welcoming trend, and the pioneering work of Pustejovsky (2012), Lou (2011), and Asher (2015) are indeed efforts in the right direction. But these approaches lack generality, as we discussed above. Moreover, these approaches are limited to one or two phenomena, but do not offer a general framework that suggests how the root problem of uncovering the missing and assumed text can be solved. The problem, in our opinion, lies in our logical formalism. In particular, we believe that formal semantics faltered due to how predication has been used to represent both, genuine predicates, as well as types, and it is precisely this subtle oversight that is at the source of many 'apparent' challenges in formal semantics.

3 Ontological vs. Logical Concepts

We begin by making the case for a very crucial aspect to our proposal, namely distinguishing between *ontological concepts* (that are **types** in a strongly-typed ontology) and *logical concepts* (that are **predicates** that represent the properties of and the relations that hold between objects of various ontological types).

3.1 Types vs. Predicates

In *Types and Ontology*, Fred Sommers (1963) suggested that there is a strongly typed ontology (that he termed 'the language tree') that seems to be implicit in all that we say in ordinary spoken language, where two objects x and y are considered to be of the same type *if and only if* (abbr.: iff) the set of monadic predicates that are significantly (that is, truly or falsely, but not absurdly) predicable of x is equivalent to the set of predicates that are significantly predicable of x. Thus, for example, while the noun phrases in (4) make reference to four distinct sets of objects, for an ontologist interested in the relationship between ontology and language, the noun phrases in (7) are ultimately referring to two types only, namely Cat and Number:

(7) a. *an old cat*
b. *a black cat*
c. *an even number*
d. *a prime number*

In other words, whether we make a reference to an *old cat* or to a *black cat*, in both instances we are ultimately speaking of objects that are of the same type; and this, according to Sommers, is a reflection of the fact that the set of monadic predicates in our natural language that are significantly predicable of OLD cats is *exactly* the same set that is significantly predicable of BLACK cats (or, whatever can sensibly be said of *black cats* can also be sensibly said of *old cats*, and vice versa). In this sense, a concept such as OLD is a **predicate** that happens to be predicable of a concept such as Cat, which corresponds to a **type** in a strongly-typed ontology.[1] As such, we take the proper logical representation for the noun phrase in (8) to be that in (8b), and not the one in (8a).

(8) $[\![\textit{an adorable cat}]\!]$
a. $\Rightarrow \lambda P[(\exists x)(\textsc{cat}(x) \wedge \textsc{adorable}(x) \wedge P(x))]$
b. $\Rightarrow \lambda P[(\exists x :: \mathsf{Cat})(\textsc{adorable}(x) \wedge P(x))]$

That is, 'an adorable cat' refers to some ADORABLE object of type Cat. Note also that abstract objects, such as events, states, properties, etc. are also types in the ontology and can also be predicated, as shown in (9).

[1] As Hacking (2001) suggests, one can think of a type such as Cat to be the kind of object that is an answer to a question such as 'What-is-it?' Thus the distinction between types and predicates might be related to Kant's analytic/synthetic distinction, where the truth of a type judgment such as ($\textit{Sheba} :: \mathsf{Cat}$) is a synthetic judgment the truth of which is determined by virtue of what we know about the world; while the truth of the judgment $\textsc{wild}(\textit{Sheba} :: \mathsf{Cat})$ is determined by virtue of the meaning of WILD; i.e. by what we take WILD to mean. As such, all entities in our system are object of a certain type, including abstract objects such as events, activities, properties, states, etc. Thus, there are no 'teachers' in our ontology, but objects of type Human that might be the agents of some TEACHING event.

(9) $[\![\mathit{an\ imminent\ event}]\!] \Rightarrow \lambda P[(\exists x :: \mathsf{Event})(\textsc{imminent}(x) \wedge P(x))]$
$[\![\mathit{an\ idle\ state}]\!] \Rightarrow \lambda P[(\exists x :: \mathsf{State})(\textsc{idle}(x) \wedge P(x))]$
$[\![\mathit{a\ desirable\ property}]\!] \Rightarrow \lambda P[(\exists x :: \mathsf{Property})(\textsc{desirable}(x) \wedge P(x))]$

In our representation, therefore, we assume a Platonic universe that includes everything we talk about in our language, and where concepts belong to two quite distinct categories: (i) ontological concepts, such as Animal, Substance, Entity, Artifact, Book, Event, State, etc., which are types in a subsumption hierarchy, and where the fact that an object of type Human is (ultimately) an object of type Entity is expressed as Human $\sqsubseteq$ Entity; and (ii) logical concepts, such as FORMER, OLD, IMMINENT, BEAUTIFUL, etc., which are the properties (that can be said) of, and the relations (that can hold) between, ontological concepts. The following are examples that illustrate the difference between logical and ontological concepts:

(10)
R_1 : OLDER(x :: Entity)
R_2 : HEAVY(x :: Physical)
R_3 : HUNGRY(x :: Living)
R_4 : ARTICULATE(x :: Human)
R_5 : MAKE(x :: Human, y :: Artifact)
R_6 : MANUFACTURE(x :: Human, y :: Instrument)
R_7 : RIDE(x :: Human, y :: Vehicle)
R_8 : DRIVE(x :: Human, y :: Car)

The predicates in (10) are supposed to reflect the fact that in ordinary spoken language we can say OLD of any Entity; that we say HEAVY of objects that are of type Physical; that HUNGRY is said of objects that are of type Living; that ARTICULATE is said of objects that must be of type Human; that MAKE is a relation that can hold between a Human and an Artifact; that MANUFACTURE is a relation that can hold between a Human and an Instrument, etc. Note that the type assignments in (10) implicitly define a type hierarchy as that shown in Figure 1. Consequently, and although not explicitly stated in (10), in ordinary spoken language one can always attribute the property HEAVY to an object of type Car since Car $\sqsubseteq$ Vehicle $\sqsubseteq$ Physical, where $\sqsubseteq$ should be read as 'subtype-of.'[2]

In addition to logical and ontological concepts, there are also proper nouns, which are the names of objects that could be of any type. A proper noun, such as *Sheba*, is interpreted as follows: $[\![\mathit{Sheba}]\!] \Rightarrow \lambda P[(\exists^1 \mathit{Sheba} :: \mathsf{Thing})\,(P\,(x))]$.

A point worth mentioning at this early juncture is that besides the embedding of 'commonsense' constraints in our predicates, what implicitly gets defined by applying Sommers' predicability test, as given by (10), is the implicit determination of 'saliency.' For example, and while it makes sense in our everyday discourse to speak of Human objects that MAKE, RIDE and DRIVE objects of type Car, DRIVE is a more salient relation between a Human and a Car, since a Human rides a car as a Vehicle, and makes a car as an Artifact, but drives a car explicitly as a Car (see Fig. 2). We will discuss this in more detail below.

[2]It should be noted here that the expressions in (10) are assumed to refer to a specific sense of each predicate. In general, however, the type assignment is a set of possible types where a single type is eventually left after lexical disambiguation. This will be discussed in more detail below.

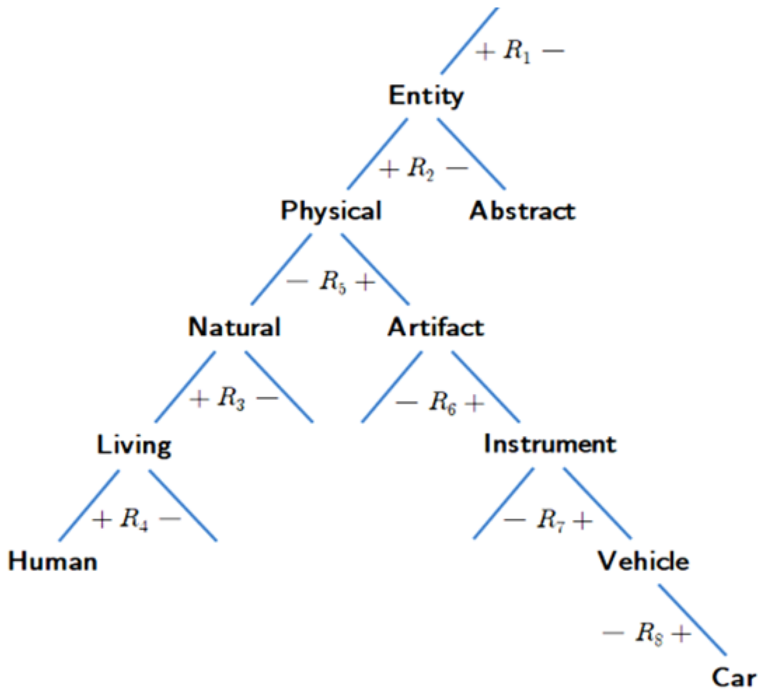

Figure 1: The type hierarchy implied by the type assignments in (10).

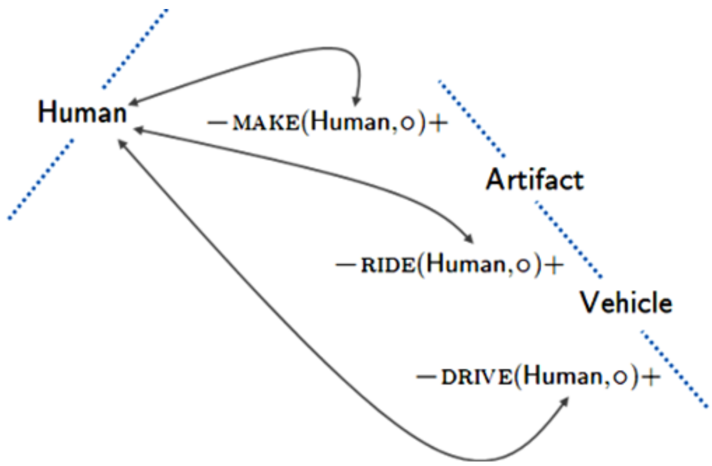

Figure 2: Salient relations implied by the ontological types and their properties.

3.2 Type Unification

Let us now start our 'compositional' semantics. Consider the interpretation of *Sheba is a thief*, where we assume THIEF is a property that is ordinarily said of objects that must be of type Human, that is $\text{THIEF}(x :: \mathsf{Human})$:

$$[\![Sheba\,is\,a\,thief]\!] \Rightarrow \lambda P[(\exists^1 Sheba :: \mathsf{Thing})(\text{THIEF}(Sheba :: \mathsf{Human}))]$$

Thus, *Sheba is a thief* is interpreted as: There is a unique object named *Sheba* (initially assumed to be of type Thing), such that the property THIEF is true of *Sheba*.[3]

Note now that in (11) *Sheba* is associated with more than one type in a single scope. In these situations a type unification must occur, where a type unification $(\mathsf{S} \bullet \mathsf{T})$ between S and T and where $Q \in \{\forall, \exists\}$, is defined for now as follows:

(11) $$Q\,(x :: (\mathsf{S} \bullet \mathsf{T}))\,(P\,(x))$$

$$\equiv \begin{cases} Q\,(x :: (\mathsf{S}))\,(P\,(x))\,, & if\ \ (\mathsf{S} \sqsubseteq \mathsf{T}) \\ Q\,(x :: (\mathsf{T}))\,(P\,(x))\,, & if\ \ (\mathsf{T} \sqsubseteq \mathsf{S}) \\ Q\,(x :: (\mathsf{S}))\,Q\,(x :: (\mathsf{T}))\,(P\,(x) \wedge \mathbf{R}\,(x,y))\,, & if\ \ (\exists \mathbf{R})\,(\mathbf{R} = msr\,(\mathsf{S},\mathsf{T})) \\ \perp, & otherwise \end{cases}$$

where $msr\,(\mathsf{S},\mathsf{T})$ stands for the *most salient relation* between objects of type S and objects of type T. That is, in situations where there is no subsumption relation between S and T the type unification results in keeping the variables of both types and in introducing some salient relation between the two types (we will discuss these situations below). Going back to (11), the type unification in this case is actually quite simple, since $(\mathsf{Human} \sqsubseteq \mathsf{Thing})$:

(12) $$\begin{aligned} & [\![Sheba\,is\,a\,thief]\!] \\ & \Rightarrow (\exists^1 Sheba :: \mathsf{Thing})(\text{THIEF}(Sheba :: \mathsf{Human})) \\ & \Rightarrow (\exists^1 Sheba :: (\mathsf{Human} \bullet \mathsf{Thing})(\text{THIEF}(Sheba)) \\ & \Rightarrow (\exists^1 Sheba :: \mathsf{Human})(\text{THIEF}(Sheba)) \end{aligned}$$

In the final analysis, therefore, *Sheba is a thief* is interpreted as follows: there is a unique object named *Sheba*, an object that eventually came out to be of type Human, such that THIEF is true of *Sheba*. Note the clear distinction between ontological concepts (e.g., Human), which Cocchiarella (2001) calls first-intension concepts, and logical (or second-intension) concepts, such as $\text{THIEF}(x :: \mathsf{Human})$. In accordance with Quine's famous slogan ("to be is to be the value of a variable"), what (12) says is that what ontologically exist are objects of type Human, and not thieves, and THIEF is an accidental (and temporal, etc.) property that we came to use to talk of certain objects of type Human. Furthermore, it is assumed that a logical concept such as THIEF is defined by a logical expression such as $(\forall x :: \mathsf{Human})(\text{THIEF}(x) \equiv \phi)$, where the exact nature of ϕ might very well be susceptible to temporal, cultural, and other contextual factors, depending on what, at a certain point in time, a certain community considers a thief to be.

[3] For simplicity, we are ignoring for now some intermediate steps in the translation, especially as it relates to the copula 'is' which plays an important part in determining the correct predication (more on this below).

It should also be noted that a first-intension such as $(x :: \mathsf{Human})$, as well as a second-intension such as $\textsc{articulate}(x :: \mathsf{Human})$, are both 'judgments' where the former is a type judgment and the latter is a value judgment. As such, in the interpretation of 'John is articulate' the first-intension $(x :: \mathsf{Human})$ must precede the second-intension $\textsc{articulate}(x :: \mathsf{Human})$, unlike the 'in-parallel' evaluation $\textsc{human}(John) \wedge \textsc{articulate}(John)$ we usually get in standard first-order logic. Another way of stating this is that for a judgment such as $\textsc{articulate}(x :: \mathsf{Human})$ to be made, the judgment $(x :: \mathsf{Human})$ must first be made. The way we see it, therefore, is that type judgments are the first level in the entire semantic structure, as suggested in Figure 3 below.

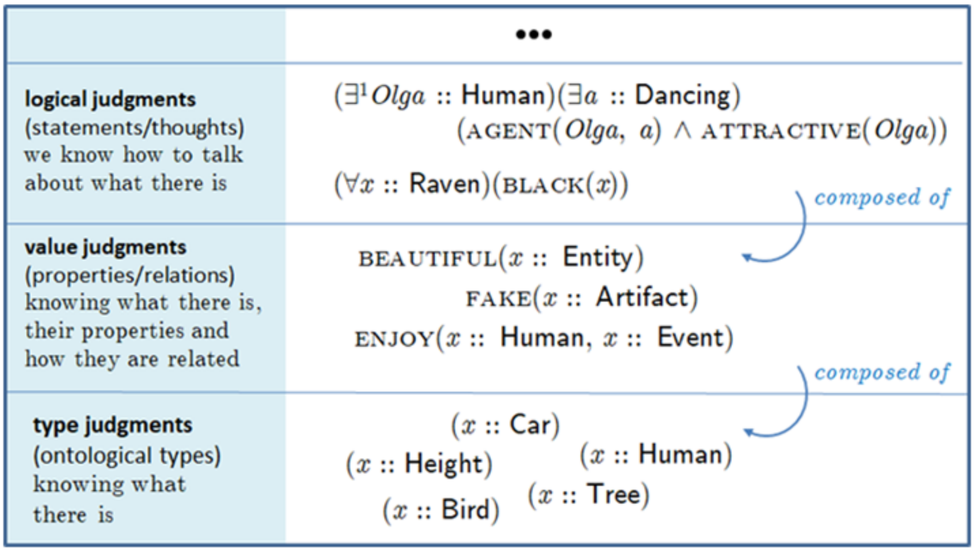

Figure 3: Levels of semantic processing.

3.3 More on Type Unification

Consider the following (initial) interpretation of *Sara owns a black cat*, where we assume $\textsc{black}(x :: \mathsf{Physical})$ and $\textsc{own}(x :: \mathsf{Human},\ y :: \mathsf{Entity})$–that is, we assume that $\textsc{black}$ can be said of objects of type $\mathsf{Physical}$, and that the $\textsc{own}$ relationship holds between objects of type Human and objects of type Entity:

(13) $[\![Sara\ owns\ a\ black\ cat]\!]$
$\Rightarrow (\exists^1 Sara :: \mathsf{Thing})(\exists c :: \mathsf{Cat})(\textsc{black}(c :: \mathsf{Physical})$
$\wedge\ \textsc{own}(Sara :: \mathsf{Human},\ c :: \mathsf{Entity}))$

Thus *Sara owns a black cat* is initially interpreted as follows: there is a unique Thing named *Sara*, and some object c of type Cat, such that c is $\textsc{black}$ (and thus in this context it must be of type $\mathsf{Physical}$), and *Sara* owns c, where in this context *Sara* must be an object of type Human and c an object of type Entity. Depending on the context they are mentioned in, therefore, *Sara* and c are assigned different types: initially considered to be a Thing, *Sara* is then considered to be an object of type

Human (when the agent of an 'owning' relationship). The object c, on other hand, is assigned three types in a single scope: introduced as an object of type Cat, it is then considered to be an object of type Physical (when predicated by BLACK) and as an Entity (when the object of an 'owning' relation). The type unifications that must occur in this situation are the following (where '$\rightarrow$' means 'unifies to'):

$(Sara :: (\mathsf{Thing} \bullet \mathsf{Human}))$
$\rightarrow (Sara :: \mathsf{Human})$

$(c :: (\mathsf{Physical} \bullet \mathsf{Entity}) \bullet \mathsf{Cat}))$
$\rightarrow (c :: (\mathsf{Physical} \bullet \mathsf{Cat}))$
$\rightarrow (c :: \mathsf{Cat})$

The final interpretation of *Sara owns a black cat* is finally given by the following:

(14) $[\![Sara\ owns\ a\ black\ cat]\!]$
$\Rightarrow (\exists^1 Sara :: \mathsf{Human})(\exists c :: \mathsf{Cat})(\textsc{black}(c) \wedge \textsc{own}(Sara,\ c))$

That is, there is a unique object named *Sara*, which is of type Human, and some object c of type Cat, where c is BLACK and *Sara* OWNS c.

3.4 Type Unification and Abstract Objects

As discussed above, logical concepts such as TEACHER, THIEF, etc. are assumed to be defined by some logical expression. A plausible (although admittedly simplistic) definition for a logical concept such as DANCER could for example be given by (15).

(15) $(\forall x :: \mathsf{Human})(\textsc{dancer}(x) \equiv \exists a :: \mathsf{Dancing})(\textsc{agent}(a, x))$

That is, any x (that must be of type Human) is a DANCER iff x is the agent of some Dancing (which is a subtype Activity). Let us now consider the interpretation of *Olga is a beautiful dancer*, where we assume $\textsc{beautiful}(a :: \mathsf{Entity})$, i.e. 'beautiful' can be said of any Entity:

(16) $[\![Olga\, is\, a\, beautiful\, dancer]\!]$
$\Rightarrow (\exists^1 Olga :: \mathsf{Thing})(\exists a :: \mathsf{Dancing})$
$(\textsc{agent}(a :: \mathsf{Activity},\ Olga :: \mathsf{Human})$
$\wedge\ (\textsc{beautiful}(Olga :: \mathsf{Entity}) \vee \textsc{beautiful}(a :: \mathsf{Entity})))$

Thus, *Olga is a beautiful dancer* is initially translated as follows: there is a unique Thing named *Olga*, and some Dancing a, where *Olga* is the agent of this dancing, which must be an Activity (and as the agent, *Olga* must be of type Human), and where either *Olga* is BEAUTIFUL or her Dancing (or of course, both) is. Note now that in the same scope *Olga* and the dancing activity a are assigned three types, triggering the following type unifications:

$(Olga :: ((\mathsf{Thing} \bullet \mathsf{Entity}) \bullet \mathsf{Human}))$
$\rightarrow (Olga :: (\mathsf{Entity} \bullet \mathsf{Human}))$
$\rightarrow (Olga :: \mathsf{Human})$

$(a :: ((\mathsf{Dancing} \bullet \mathsf{Activity}) \bullet \mathsf{Entity}))$
$\rightarrow (a :: (\mathsf{Dancing} \bullet \mathsf{Entity}))$
$\rightarrow (a :: \mathsf{Dancing})$

Concerning the disjunction term in (16), representing the ambiguity in nominal modification, we now have the following $\textsc{beautiful}(\textit{Olga} :: \mathsf{Human}) \vee \textsc{beautiful}(a :: \mathsf{Dancing}))$. Since both terms in the disjunction are acceptable, the final translation is the one given in (17) that admits an ambiguity in nominal modification, i.e. the possibility of 'beautiful' describing *Olga* or her dancing (or both).

(17) $[\![Olga\ is\ a\ beautiful\ dancer]\!]$
$\Rightarrow (\exists^1 \textit{Olga} :: \mathsf{Human})(\exists a :: \mathsf{Dancing})$
$(\textsc{agent}(a, \textit{Olga}) \wedge (\textsc{beautiful}(\textit{Olga}) \vee \textsc{beautiful}(a)))$

Unlike the situation in (13), however, the relevant type unifications should remove the ambiguities in (18) and (19),

(18) *Olga is an experienced dancer*

(19) *Olga is a recreational dancer*

where it is clear that *experienced* is describing *Olga* in the former and *recreational* is describing *Olga*'s dancing in the latter (that is, in (14) it is *Olga* and not her dancing that is 'experienced', and in (15) it is not *Olga*, but her dancing that is 'recreational'). The only term we need to consider here is the term involving the disjunction representing the ambiguity in nominal modification. Here are the type unifications in the case of (18):

$\textsc{experienced}(\textit{Olga} :: (\mathsf{Human} \bullet \mathsf{Human})) \vee \textsc{experienced}(a :: (\mathsf{Dancing} \bullet \mathsf{Human}))$
$\rightarrow \textsc{experienced}(\textit{Olga} :: \mathsf{Human}) \vee \textsc{experienced}(a :: \bot)$
$\rightarrow \textsc{experienced}(\textit{Olga} :: \mathsf{Human}) \vee \bot$
$\rightarrow \textsc{experienced}(\textit{Olga} :: \mathsf{Human})$

The type unification admitting an 'experienced dancing' fails here, leaving 'experienced' to unambiguously modify *Olga*. In (19), however, we have the following:

$\textsc{recreational}(\textit{Olga} :: (\mathsf{Human} \bullet \mathsf{Dancing}))$
$\vee\ \textsc{recreational}(a :: (\mathsf{Dancing} \bullet \mathsf{Dancing}))$
$\rightarrow \textsc{recreational}(\textit{Olga} :: \bot) \vee \textsc{recreational}(a :: \mathsf{Dancing})$
$\rightarrow \bot \vee \textsc{recreational}(a :: \mathsf{Dancing})$
$\rightarrow \textsc{recreational}(a :: \mathsf{Dancing})$

Note that in this case the type unification removes 'recreational *Olga*' leaving 'recreational' to unambiguously modify *Olga*'s dancing. One might at this point question why the type unification ($\mathsf{Dancing} \bullet \mathsf{Human}$) in (18) and (19) was considered a failure (resulting in $\bot$), although the definition of type unification given in (11) suggests that in the absence of a subsumption relation between two types S and T, an attempt is first made to pick-up the most salient relation (*msr*) between the two ontological types. The answer is that looking for an *msr* occurs when all else fails, while this is not the case in (18) and (19), where the local context provided a successful type unification and thus looking elsewhere to 'make sense' of what is being said is not needed!

3.5 Failed Terms as Failed Type Unifications

Recall the interpretation of *Olga is a beautiful dancer* in (16), where the final interpretation admitted the ambiguity in nominal modification, since all type unifications succeeded, allowing 'beautiful' to remain ambiguous in that it could be modifying *Olga* or her Dancing. Let us now consider the interpretation of *Olga is a beautiful tall dancer*, where we assume $\textsc{tall}(x :: \mathsf{Physical})$:

(20) $$\begin{aligned}&[\![Olga\ is\ a\ beautiful\ tall\ dancer]\!]\\ &\Rightarrow (\exists^1 Olga :: \mathsf{Thing})(\exists a :: \mathsf{Dancing})\\ &\qquad (\textsc{agent}(a :: \mathsf{Activity},\ Olga :: \mathsf{Human})\\ &\qquad\qquad \wedge\ (\textsc{beautiful}(\textsc{tall}(Olga :: \mathsf{Physical}) :: \mathsf{Entity})\\ &\qquad\qquad\ \vee\ (\textsc{beautiful}(\textsc{tall}(a :: \mathsf{Physical}) :: \mathsf{Entity})))\end{aligned}$$

The type unifications required here are the following:

$$\begin{aligned}&(Olga :: ((\mathsf{Thing} \bullet \mathsf{Physical}) \bullet \mathsf{Human}))\\ &\to (Olga :: (\mathsf{Physical} \bullet \mathsf{Human}))\\ &\to (Olga :: \mathsf{Human})\end{aligned}$$

$$\begin{aligned}&(a :: ((\mathsf{Dancing} \bullet \mathsf{Activity}) \bullet \mathsf{Physical}))\\ &\to (a :: (\mathsf{Dancing} \bullet \mathsf{Physical}))\\ &\to (a :: \bot)\\ &\to \bot\end{aligned}$$

resulting in (21):

(21) $$\begin{aligned}&[\![Olga\ is\ a\ beautiful\ tall\ dancer]\!]\\ &\Rightarrow (\exists^1 Olga :: \mathsf{Human})(\exists a :: \mathsf{Dancing})\\ &\qquad (\textsc{agent}(a,\ Olga)\\ &\qquad\qquad \wedge\ (\textsc{beautiful}(\textsc{tall}(Olga :: \mathsf{Human}))\\ &\qquad\qquad\quad \vee\ (\textsc{beautiful}(\textsc{tall}(a :: \bot) :: \mathsf{Entity})))\\ &\Rightarrow (\exists^1 Olga :: \mathsf{Human})(\exists a :: \mathsf{Dancing})(\textsc{agent}(a,\ Olga)\\ &\qquad \wedge\ (\textsc{beautiful}(\textsc{tall}(Olga)) \vee \bot))\\ &\Rightarrow (\exists^1 Olga :: \mathsf{Human})(\exists a :: \mathsf{Dancing})(\textsc{agent}(a,\ Olga)\\ &\qquad \wedge\ (\textsc{beautiful}(\textsc{tall}(Olga)))\end{aligned}$$

Unlike the situation in (16), where 'beautiful' could be describing *Olga* or her dancing, the situation in (17) is quite different due to the adjective 'tall' that forced 'beautiful' to be describing a physical object, and thus *Olga*, and not her Dancing.

We leave it to the reader to work out why (and how) *Olga is a beautiful tall dancer* sounds fine, while *Olga is a tall beautiful dancer* sounds awkward (hint: in the latter, the type castings and type unifications reduce the entire disjunction to $\bot$!)

3.6 What 'Paradox of the Ravens'?

Before we continue showing the utility of our framework, we briefly detour to show how our ontological semantics can help us solve some longstanding logical/semantic puzzles.

Introduced in the 1940's by Carl Gustav Hempel, then a student of Carnap, the Paradox of the Ravens–or Hempel's Paradox, or the Paradox of Confirmation (Hempel, 1945)–has continued to occupy logicians, statisticians, and philosophers of science to this day. The paradox arises when one considers what constitutes an evidence for a statement (hypothesis) and it can be described by the following:

(22) H1: *All ravens are black*
H2: *All non-black things are not ravens*

Hypothesis H1 is logically equivalent to hypothesis H2, as is typically shown in the following translation in standard first-order predicate logic:

(23) H1: $(\forall x)(\text{RAVEN}(x) \supset \text{BLACK}(x))$
H2: $(\forall x)(\neg\text{BLACK}(x) \supset \neg\text{RAVEN}(x))$

The assumed 'paradox' is now due to the irreconcilability of the following, separately valid claims: (i) any observation of an instance that satisfies H1 is said to confirm H1 (to some degree); and (ii) any observation that confirms a hypothesis H1 must confirm (and to the same degree) a hypothesis that is logically equivalent to it; and (iii) the observation of any non-black thing that is also not a raven confirms H2, and thus, it confirms the logically equivalent hypothesis H1. This, however, leaves us with the unpleasant conclusion that observing a white shoe, a blue shirt, or, for that matter, any non-black and non-raven object currently in your sight, confirming 'all ravens are black.' Clearly, this cannot be accepted, since the observation of a red herring, say, has no bearing on the hypothesis that 'all ravens are black.'

Many solutions have been proposed to the Paradox of the Ravens that range from accepting the paradox (that observing red apples and other non-black non-ravens does confirm the hypothesis 'All ravens are black'–since the observation *does not disconfirm* it), to proposals in the Bayesian tradition that try to measure the 'degree' of confirmation. Concerning the latter, Bayesian proposals essentially amount to proposing that observing a red apple, for example, does confirm the hypothesis 'All ravens are black' but it does so very minimally, and certainly much less than the observation of a black raven (see Vranas, 2004). Clearly, however, this is not a satisfactory solution, since observing a red flower should not contribute at all to the confirmation of 'All ravens are black.' Worse yet, it turns out that in the Bayesian analysis the observation of black but non-raven objects actually negatively confirms the hypothesis that 'All ravens are black,' which is also problematic. An even more serious flaw in the Bayesian analysis is this: if observing a red apple confirms the hypothesis that 'All ravens are black' (no matter how minutely), then it also minimally, but equally, confirms the hypothesis that 'All ravens are green.' Thus, we have the same observation that confirms two mutually exclusive hypotheses, which clearly is unacceptable (see Maher, 1999, for a more detailed discussion).

What we suggest here is that the purported 'paradox' is not a logical one, but a representational one. In particular, we believe that the problem lies in the standard first-order predicate logic formulation of the hypothesis and its contrapositive. Let us suggest embedding ontological types in the standard Montague (1974) representation, where we are assuming $\text{BLACK}(x :: \mathsf{Physical})$, i.e. BLACK is a property that is ordinarily said of objects of type $\mathsf{Physical}$.

We will start first with H1, namely the hypothesis *All ravens are black.* Note the crucial step (24.3), where unifying the term $(x :: \mathsf{Raven})$ with the variable introduced by the quantifier reduces the term $(x :: \mathsf{Raven})$ to **true**, and thus the meaning of *All ravens* to $\lambda Q[(\forall x :: \mathsf{Raven})(Q(x))]$. In (24.7), the term $\text{BLACK}(x :: \mathsf{Physical})$ forces a type unification $(\mathsf{Raven} \bullet \mathsf{Physical})$, resulting in the final interpretation of *All ravens are black* as $(\forall x :: \mathsf{Raven})(\text{BLACK}(x))$–that is, for any object x of type Raven, BLACK is true of x.

(24)
1. $[\![All]\!] \Rightarrow \lambda P \lambda Q\,[(\forall x :: \mathsf{Thing})\,(P\,(x) \supset Q\,(x))]$
2. $[\![ravens]\!] \Rightarrow \lambda x\,[(x :: \mathsf{Raven})]$
3. $[\![All\ ravens]\!] \Rightarrow \lambda Q\,[(\forall x :: \mathsf{Thing})\,((x :: \mathsf{Raven}) \supset Q\,(x))]$
$\Rightarrow \lambda Q\,[(\forall x :: (\mathsf{Thing} \bullet \mathsf{Raven}))\,((x :: \mathsf{Raven}) \supset Q\,(x))]$
$\Rightarrow \lambda Q\,[(\forall x :: \mathsf{Raven})\,((x :: \mathsf{Raven}) \supset Q\,(x))]$
$\Rightarrow \lambda Q\,[(\forall x :: \mathsf{Raven})\,((x :: \mathbf{true}) \supset Q\,(x))]$
$\Rightarrow \lambda Q\,[(\forall x :: \mathsf{Raven})\,(Q\,(x))]$
4. $[\![are]\!] \Rightarrow \lambda x\,[P\,(x :: \mathsf{Thing})]$
5. $[\![black]\!] \Rightarrow \lambda x[(\textsc{black}(x :: \mathsf{Physical}))]$
6. $[\![are\ black]\!] \Rightarrow \lambda x[(\textsc{black}(x :: \mathsf{Physical}))]$
7. $[\![All\ ravens\ are\ black]\!] \Rightarrow (\forall x :: \mathsf{Raven})(\textsc{black}(x :: \mathsf{Physical}))$
$\Rightarrow (\forall x :: (\mathsf{Raven} \bullet \mathsf{Physical}))(\textsc{black}(x))$
$\Rightarrow (\forall x :: \mathsf{Raven})(\textsc{black}(x))$

Let us now consider the logically equivalent hypothesis H2, namely that *All non-black things are not ravens*:

(25)
1. $[\![All]\!] \Rightarrow \lambda P \lambda Q\,[(\forall x :: \mathsf{Thing})\,(P\,(x) \supset Q\,(x))]$
2. $[\![non-black]\!] \Rightarrow \lambda x[(\neg\textsc{black}(x :: \mathsf{Physical})]$
3. $[\![All\ non-black]\!] \Rightarrow \lambda Q[(\forall x :: \mathsf{Thing})(\neg\textsc{black}(x :: \mathsf{Physical}) \supset Q\,(x))]$
$\Rightarrow \lambda Q[(\forall x :: (\mathsf{Thing} \bullet \mathsf{Physical}))(\neg\textsc{black}(x :: \mathsf{Physical}) \supset Q\,(x))]$
$\Rightarrow \lambda Q[(\forall x :: \mathsf{Physical})(\neg\textsc{black}(x :: \mathsf{Physical}) \supset Q\,(x))]$
$\Rightarrow \lambda Q[(\forall x :: \mathsf{Physical})(\neg\textsc{black}(x) \supset Q\,(x))]$
4. $[\![are]\!] \Rightarrow \lambda P \lambda x\,[P\,(x :: \mathsf{Thing})]$
5. $[\![non-ravens]\!] \Rightarrow \lambda x[(\neg(x :: \mathsf{Raven}))]$
6. $[\![are\ non-ravens]\!] \Rightarrow \lambda x[(\neg(x :: \mathsf{Raven}))]$
7. $[\![All\ non-black\ are\ non-ravens]\!]$
$\Rightarrow (\forall x :: \mathsf{Physical})(\neg\textsc{black}(x)) \supset \neg(x :: \mathsf{Raven}))$
$\Rightarrow (\forall x :: (\mathsf{Raven} \bullet \mathsf{Physical}))(\neg\textsc{black}(x)) \supset \neg(x :: \mathsf{Raven}))$
$\Rightarrow (\forall x :: \mathsf{Raven})(\neg\textsc{black}(x) \supset \neg(x :: \mathsf{Raven}))$
$\Rightarrow (\forall x :: \mathsf{Raven})(\neg\textsc{black}(x) \supset \neg\mathbf{true})$
$\Rightarrow (\forall x :: \mathsf{Raven})(\mathbf{false} \vee \textsc{black}(x))$
$\Rightarrow (\forall x :: \mathsf{Raven})(\textsc{black}(x))$

Note that in this situation the widest scope variable x remains to be of type Physical until the last step (25.7) where the unification (Raven • Physical) casts x to a specific Physical object, namely Raven. What matters for us here is that in a formal semantics embedded with types from a strongly-typed ontology both *All ravens are black* and its logically equivalent *All non-black things are not ravens* turn out to be equivalent not only in content, but also in form, where the logical formulation of both is $(\forall x :: \mathsf{Raven})(\textsc{black}(x))$, and thus there is no 'paradox' of the ravens since the logically equivalent hypotheses H1 and H2 are now confirmed (and disconfirmed) by the same observations!

4 Where Formal Semantics Can (Should?) Go

In this section we show how the embedding of ontological concepts in our predicates can help us tackle some well-known challenges in the semantics of natural language.

4.1 Lexical Disambiguation

For now we have been assuming single type assignments to the variables of our predicates, e.g. BLACK(x :: Physical), BEAUTIFUL(x :: Entity), etc., as it was implicitly assumed that the predicate in question has been disambiguated–that is, that a specific meaning of the predicate has been selected. We will continue to do so where the context is clear, although we will show now how lexical disambiguation itself is conducted in our system, and that requires that we initially consider, for some terms, a set of type assignments.

Let us consider the interpretation of the sentence in (26), where we will concentrate on two senses of 'party,' i.e. we will assume that 'party' belongs to (at least) two branches in our ontological structure, as shown in Figure 4.

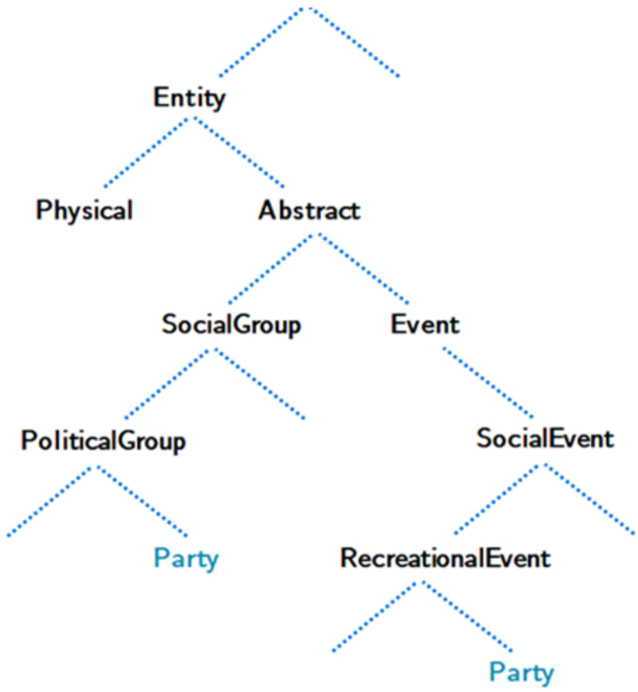

Figure 4: Lexical ambiguity as multiple ontological types.

(26) $[\![\textit{Jon cancelled the party}]\!]$
$\Rightarrow (\exists^1 Jon :: \mathsf{Thing})(\exists a :: \mathsf{Activity})(\exists^1 p :: \{\mathsf{PoliticalGroup},\ \mathsf{RecreationalEvent}\})$
$(\textsc{cancellation}(a) \land \textsc{subject}(a, Jon :: \mathsf{Human})$
$\land\ \textsc{object}(a, p :: \mathsf{Event}))$

Thus, initially, the types associated with 'the party' p is a set of all possible types (again, for simplicity we assumed that 'cancelled' has been disambiguated where it was determined that it is some activity the object of which is an object of type Event). The type unifications concerning *Jon* are straightforward. The type unifications that must occur for p are now a set of n pairs of type unifications, where n is the number of possible meanings of p:

$(p :: \{(\mathsf{Event} \bullet \mathsf{PoliticalGroup}),(\mathsf{Event} \bullet \mathsf{RecreationalEvent})\})$
$(p :: \{\perp,\mathsf{RecreationalEvent}\})$
$(p :: \{\mathsf{RecreationalEvent}\})$

Thus, the initial set of types is reduced to a singleton and the 'party' that John seems to have cancelled is the 'recreational event' meaning of 'party.' Note that if 'cancelled' in (26) was replaced by 'assisted,' where it is assumed that the object of 'assistance' can be a human or a social group, then the correct meaning of 'party' will also be selected, namely the meaning of the social (political) group. On the other hand, if 'cancelled' where to be replaced by 'promoted,' then we would have a genuinely ambiguous statement, since one can 'promote' a political group, as well as a recreational event, as illustrated by (27).

(27) $[\![Jon\ promoted\ the\ party]\!]$
$\Rightarrow (\exists^1 Jon :: \mathsf{Thing})(\exists a :: \mathsf{Activity})(\exists^1 p :: \{\mathsf{PoliticalGroup},\ \mathsf{RecreationalEvent}\})$
$(\text{PROMOTION}(a) \wedge \text{SUBJECT}(a,\ Jon :: \mathsf{Human})$
$\wedge\ \text{OBJECT}(a,\ p :: \mathsf{Entity}))$

Assuming that any Entity can be promoted, and in the absence of any additional information (and thus additional type constraints), both meanings of 'party' remain to be equally plausible.

4.2 Metonomy, Copredication, and Salient Meanings

Consider the sentence in (28), where two senses of 'book' are assumed to be used in the same context, the informational content sense of book (when being read) and the physical object sense (when being burned):

(28) *Jon read the book and then he burned it.*

In Asher and Pustejovsky (2005), it is argued that this is an example of what they term copredication; where incompatible predicates are applied to the same type of object. It is argued that in (29), for example, 'book' must have what is called a dot type, which is a structured object that in a sense carries the 'informational content' sense (which is referenced when it is being read), as well as the 'physical object' sense (which is referenced when it is being burned). Elaborate machinery is then introduced to 'pick out' the right sense in the right context, and all in a well-typed compositional logic. But this approach presupposes that one can enumerate, *a priori*, all possible uses of the word 'book' in ordinary language. More to the point, we believe that what is termed 'copredication' is not different, in essence, from what is known as metonymy, as in both cases we are trying to either 'pick-up' (i) a hidden essence/sense of some object; or (ii) some hidden (i.e. unstated) relation between some objects in the discourse under consideration. To illustrate this point further, let us first consider the interpretation of *Jon bought and studied Das Kapital*:

(29) $[\![Jon\ bought\ and\ studied\ Das\ Kapital]\!]$
$\Rightarrow (\exists^1 Jon :: \mathsf{Thing})(\exists^1 DasKapital :: \mathsf{Book})(\exists a_1 :: \mathsf{Activity})(\exists a_2 :: \mathsf{Activity})$
$(\text{STUDYING}(a_1) \wedge \text{SUBJECT}(a,\ Jon :: \mathsf{Human})$
$\wedge\ \text{OBJECT}(a_1,\ DasKapital :: \mathsf{InformationalMaterial})) \wedge$
$(\text{BUYING}(a_2) \wedge \text{SUBJECT}(a,\ Jon :: \mathsf{Human})$
$\wedge\ \text{OBJECT}(a_1,\ DasKapital :: \mathsf{Physical}))$

That is, *Jon* bought a book that he also studied, where the object of the BUYING activity must be an object of type Physical, and the object of a STUDYING is some InformationalMaterial. The type unifications of *Jon* and the (object of BUYING) Physical book *DasKapital* are straightforward:

$(Jon :: (\mathsf{Human} \bullet \mathsf{Thing})) \rightarrow (Jon :: \mathsf{Thing})$
$(DasKapital :: (\mathsf{Physical} \bullet \mathsf{Book})) \rightarrow (DasKapital :: \mathsf{Book})$

However, the object of the STUDYING activity, which is a Book, must be an InformationalMaterial object when being read. For these two types unify the most salient relation between Book and InformationalMaterial that is picked-up, introducing in the process a new variable *infoc* that is related to a book: HASCONTENT(*DasKapital*, *infoc*). The final translation is thus that given by (30).

(30) $$\begin{aligned}&[\![Jon\ bought\ and\ studied\ Das\ Kapital]\!]\\ &\Rightarrow (\exists^1 Jon :: \mathsf{Human})(\exists^1 DasKapital :: \mathsf{Book})\\ &\quad(\exists a_1 :: \mathsf{Activity})(\exists a_2 :: \mathsf{Activity})\ (\exists infoc :: \mathsf{InformationalMaterial})\\ &\qquad(\textsc{studying}(a_1) \wedge \textsc{subject}(a,\ Jon) \wedge \textsc{object}(a,\ DasKapital)\\ &\qquad\wedge (\textsc{buying}(a_2) \wedge \textsc{subject}(a,\ Jon) \wedge \textsc{object}(a,\ DasKapital)\\ &\qquad\quad\wedge \textsc{hasContent}(DasKapital,\ infoc))\end{aligned}$$

That is, *Jon bought and studied Das Kapital* describes a situation where there is a unique object named *Jon*, an object of type Human, and some Book titled *DasKapital* (that *Jon* bought), and where *DasKapital* has the InformationalMaterial *infoc* (that *Jon* studied).

It is important to note at this stage that hidden (and implicitly assumed) information is either obtained by straightforward type unification or, when all attempts fail, by picking up some salient property or relation between the objects in the discourse. On the other hand, unwanted meanings (as in lexical disambiguation, or removing some ambiguities related to nominal modification) are obtained when certain type unifications fail.

Still with respect to the important point that we like to make here that copredication, the name given for the phenomenon exemplified by (31), is not much different from what is known by metonymy, in that type unification is the process by which an indirect reference or some salient relation are discovered, consider for example the following:

(31) $$\begin{aligned}&[\![The\ omelet\ wants\ a\ beer]\!]\\ &\Rightarrow (\exists^1 oml :: \mathsf{Omelet})(\exists b :: \mathsf{Beer})(\exists e :: \mathsf{Event})\\ &\quad(\textsc{wanting}(e) \wedge \textsc{subject}(e,\ oml :: \mathsf{Human}) \wedge \textsc{object}(e,\ b :: \mathsf{Thing}))\end{aligned}$$

In this case, resolving the situation of 'what is wanted' is quite simple: the object of wanting is an object of type Thing, and more specifically, a Beer, which works very well. However, it is the subject of the wanting that seems to be the problem: 'want' expects a Human subject, but we found an object of type Omelet. Clearly, these two types must somehow be reconciled. Since no subsumption relation exists between these types, the only way they can be reconciled is by finding some salient relation between them. As it turns out, there is a salient relationship between a Human and Food (a supertype of Omelet), namely the EAT relation, that will necessarily introduce an (implicit) object of type Human. Thus,

(32) $[\![$*The omelet wants a beer*$]\!]$
$\Rightarrow (\exists^1 oml :: \mathsf{Omelet})(\exists b :: \mathsf{Beer})(\exists e :: \mathsf{Event})(\exists a :: \mathsf{Activity})(\exists p :: \mathsf{Human})$
$(\textsc{eating}(a) \wedge \textsc{subject}(a,\ p) \wedge \textsc{object}(a,\ oml)$
$(\textsc{wanting}(e) \wedge \textsc{subject}(e,\ p) \wedge \textsc{object}(e,\ b))$

4.3 Ontological Types and the Copula

It is widely accepted in formal semantics that there are two senses for the copula 'is' in natural language. The two senses, generally attributed to Frege, are the 'is of identity,' exemplified by (33) and the 'is of predication,' exemplified by (34) (see Mendelsohn, 1987).

(33) *Billie the Kid is William H. Bonney.*

(34) *Billie the Kid is an outlaw.*

While (Mendelsohn, 1987) rejects the notion of two senses of 'is,' and correctly so, in our opinion, his treatment of 'is' boils down to considering 'identity' as a special case of predication, where uniqueness of identity is guaranteed by some special axioms to the theory. Ironically, Mendelsohn's proposal was inspired by the work of Sommers (1969), whose ideas on types and ontology (Sommers, 1963) are very much in line with the proposal we make here.

However, we believe that those proposals are lacking in that the ambiguity is not due to the flexibility of the copula 'is,' but in that fact that this copula can be highly polymorphic and can therefore trigger different interpretations depending on the types of objects that are flanked on both sides. To be sure, even in the case of (proper) names, there is no 'real' identity, except in the vacuous case of 'x is x' since $7 + 9$, is not exactly 'identical' to $\sqrt{256}$, for example, as illustrated by (35).

(35) *Mary taught her little brother that* $7 + 9 = 16$.
$\not\approx$ *Mary taught her little brother that* $7 + 9 = \sqrt{256}$.

What matters to us here is to demonstrate how the copula 'is' is treated in our system. For us, 'is' is a polymorphic function that has the general template $\textsc{is}(x :: \mathsf{S},\ y :: \mathsf{T})$ where the final interpretation is a function of the types S and T and their type unification. That is, in cases where there is no subsumption relationship between S and T, the copula 'is' will introduce a salient relation between the two types. In (36) we have some common examples that involve very general and abstract types, where $\mathbf{HasProp}(x, y)$ means x has the property y, $\mathbf{INPro}(x, y)$ means x is in (or is going through) the process y, $\mathbf{DOES}(x, y)$ means x (often) does y, and $\mathbf{INSt}(x, y)$ means x is in the state y.

For example, (36a) says that *Liz* has the property of FAME, while (36c) says that the property of FAME is desirable, etc.

(36) a. *Liz is famous*
$\Rightarrow (\exists^1 Liz :: \mathsf{Human})(\exists p :: \mathsf{Property})(\textsc{fame}(p) \wedge \mathbf{HasProp}(Liz,\ p))$

b. *Jon is aging*
$\Rightarrow (\exists^1 Jon :: \mathsf{Human})(\exists p :: \mathsf{Process})(\textsc{aging}(p) \wedge \mathbf{InPro}(Jon,\ p))$

c. *Aging is inevitable*
$\Rightarrow(\forall p :: \mathsf{Process})(\textsc{aging}(p) \supset$
$(\exists p :: \mathsf{Property})(\textsc{inevitability}(e) \wedge \mathbf{HasProp}(p,\ e)))$

d. *Fame is desirable*
$\Rightarrow(\forall f :: \mathsf{Process})(\textsc{fame}(p) \supset$
$(\exists d :: \mathsf{Property})(\textsc{desirability}(d) \wedge \mathbf{HasProp}(f,\ d)))$

e. *Olga is a dancer*
$\Rightarrow (\exists^1 Olga :: \mathsf{Human})(\exists a :: \mathsf{Activity})(\textsc{dancing}(a) \wedge \mathbf{DOES}(Olga,\ a))$

f. *Sheba is dead*
$\Rightarrow (\exists^1 Sheba :: \mathsf{Human})(\exists s :: \mathsf{State})(\textsc{death}(s) \wedge \mathbf{INSt}(Sheba,\ s))$

5 On the Nature of Ontological Structure

Throughout this paper we have assumed the existence of some ontological structure, an ontological structure the types of which are assumed to be embedded in predicates (the properties of and the relations between objects of various types). However, a valid question that one might ask is the following: how does one arrive at this ontological structure that implicitly underlies all that we say in everyday discourse? One plausible answer is the (*seemingly* circular) suggestion that the semantic analysis of natural language should itself be used to uncover this structure. In this regard, we strongly agree with Dummett (1991) who states:

> We must not try to resolve the metaphysical questions first, and then construct a meaning-theory in light of the answers. We should investigate how our language actually functions, and how we can construct a workable systematic description of how it functions; the answers to those questions will then determine the answers to the metaphysical ones.

What this suggests, and correctly so, in our opinion, is that in our effort to understand the complex and intimate relationship between ordinary language and everyday (commonsense) knowledge, one could, as Bateman (1995) has also suggested, "use language as a tool for uncovering the semiotic ontology of commonsense," since language is the only theory we have of everyday knowledge. To alleviate this seeming circularity (in assuming this ontological structure in our semantic analysis; while at the same time suggesting that semantic analysis of language should itself be used to uncover this ontological structure), we suggest performing semantic analysis from the ground up, assuming a minimal (almost a trivial and basic) ontology, building up the ontology as we go guided by the results of the semantic analysis.

The advantages of this approach are: (i) the ontology thus constructed as a result of this process would not be invented, as is the case in most approaches to ontology (e.g., Guarino, 1995; Lenat & Guha, 1990; Sowa, 1995), but would instead be discovered from what is in fact implicitly assumed in our use of language in everyday discourse; (ii) the semantics of several natural language phenomena should as a result become trivial, since the semantic analysis was itself the source of the underlying knowledge structures (in a sense, one could say that the semantics would have been done before we even started!)

Another promising technique that can be used to 'boot-up' this ontological structure is performing some corpus analysis to initially obtain sets of monadic predicates that seem to be (sensibly) used in the predication of some very general types (e.g., Artifact, Event, Physical, State, etc.). A subset relationship analysis can then be used to discover the hidden hierarchical structure. A similar approach has been suggested in Saba (2006).

6 Concluding Remarks

Most of the challenges in the semantics of natural language seem to be related to the phenomenon of the 'missing text'–that is, text that is almost never explicitly stated but is implicitly assumed as 'shared' background knowledge. As Hobbs (1985) has suggested, however, challenges in the semantic analysis of natural language can become more tractable if our formal semantics was embedded with some theory of the commonsense world, at least as we talk about it in our everyday discourse (commonsense knowledge).

In this paper, we suggested one such method, a method that rectifies what we believe was an oversight in formal semantics, namely distinguishing between two fundamentally different types of concepts: (i) ontological concepts, that correspond to types in a strongly-typed ontology; and (ii) logical concepts, that correspond to the properties of and the relations between objects of various ontological types. One method to distinguish between the two was vaguely suggested by early work in the metaphysics of natural languages (Sommers, 1963), and is also related to some very powerful ideas put forward by Cocchiarella (2001), where he makes a strong argument for a new 'logic as language'–a logic that has ontological content where one can clearly distinguish between what he calls first-intension and second-intension concepts. We have tested our approach on a number of challenges in semantics, where we suggested clear and uniform treatment of a number of phenomena–phenomena that are usually treated by proposing often incompatible solutions.

Several linguistic phenomena, some of which have been relegated to intensionality (e.g., intensional verbs), or to reasoning with abstract objects (such as '*Jon is wise*' should follow from '*exercising is wise*' and '*Jon is exercising*'), or to compound nominals, quantifier scope ambiguity, etc., were not dealt with in this paper, as that would require some minor extension to our formalism and would thus extend the paper considerably. These and a more detailed presentation of some of the subjects covered in this paper are forthcoming. Of course, much is left to be done, refined, and formally defined, and mainly the 'discovery' of that ontological structure that seems to be implicit in everything we say in everyday discourse.

Acknowledgments

The valuable feedback of Bill DeSmedt, Romeo Issa, and Luis M. Augusto are greatly appreciated. Of course, all remaining shortcomings are ours.

References

Asher, N. (2008). A type driven theory of predication with complex types. *Fundamenta Informaticae*, *84*(2), 151-183.

Asher, N. (2011). *Lexical meaning in context: A web of words.* Cambridge, etc.: Cambridge University Press.

Asher, N. (2015). Types, meanings and coercions in lexical semantics. *Lingua*, *157*, 66-82.

Asher, N. & Pustejovsky, J. (2005). Word meaning and commonsense metaphysics. Available at the semanticsarchive.net.

Asher, N. & Pustejovsky, J. (2012). A type composition logic for generative lexicon. In J. Pustejovsky, P. Bouillon, H. Isahara, K. Kanzaki, & C. Lee (eds.), *Advances in generative lexicon theory* (pp. 39-66). Dordrecht, etc.: Springer.

Bateman, J. A. (1995). On the relationship between ontology construction and natural language: A socio-semiotic view. *International Journal of Human-Computer Studies*, *43*, 929-944.

Cocchiarella, N. B. (2001). Logic and ontology. *Axiomathes*, *12*, 117-150.

Dummett. M. (1991). *Logical basis of metaphysics.* London: Duckworth.

Guarino, N. (1995). Formal ontology in conceptual analysis and knowledge representation. *International Journal of Human-Computer Studies*, *43*(5-6), 625-640.

Hacking, I. (2001). Aristotelian categories and cognitive domains. *Synthese*, *126*, 473–515.

Hempel, G. (1945). Studies in the logic of confirmation. *Mind*, *54*(213), 1-26.

Hobbs, J. (1985). Ontological promiscuity. In *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics*, 61-69.

Kirby, S., Tamariz, M., Cornish, H., & Smith, K. (2015). Compression and communication in the cultural evolution of linguistic structure. *Cognition*, *141*, 87-102.

Lenat, D. B. & Guha, R. V. (1990). *Building large knowledge-based systems: Representation and inference in the CYC Project.* Boston: Addison-Wesley.

Levesque, H. J. (2011). The Winograd Schema Challenge. In *Logical formalizations of commonsense reasoning: Papers from the AAAI 2011 Spring Symposium.* Report SS-11-06, 63-68.

Luo, Z. (2010). Type-theoretical semantics with coercive subtyping. In *Semantics and linguistic theory: Proceedings of SALT*, *20*, 38-56.

Luo, Z. (2011). Contextual analysis of word meanings in type-theoretical semantics. In S. Pogodalla & J. P. Prost (eds.), *Logical aspects of computational linguistics. LACL 2011.* Berlin, Heidelberg: Springer.

Luo, Z. (2012). Formal semantics in modern type theories with coercive subtyping. *Linguistics and Philosophy*, *35*, 491–513.

Maher, P. (1999). Inductive logic and the ravens paradox. *Philosophical Science*, *66*(1), 50-70.

Martin-Löf, P. (1984). *Intuitionistic type theory.* Napoli: Bibliopolis.

Mendelsohn, R. L. (1987). Frege's two senses of 'is'. *Notre Dame Journal of Formal Logic*, *28*(1), 139-160.

Montague, R. (1974). *Formal philosophy: Selected papers of Richard Montague.* Edited and with an introduction by R. H. Thomason. New Haven and London: Yale University Press.

Pustejovsky, J. (1995). *The Generative Lexicon.* Cambridge, MA: The MIT Press.

Saba, W. (2006). Language, logic and ontology: Uncovering the structure of commonsense knowledge. *International Journal of Human-Machine Studies*, *65*(7), 610-623.

Sommers, F. (1963). Types and ontology. *Philosophical Review*, *72*(3), 327-363.

Sommers, F. (1969). Do we need identity? *Journal of Philosophy*, *LXVI*, 499-504.

Sowa, J. F. (1995). *Knowledge representation: Logical, philosophical & computational foundations.* Boston: PWS Publishing Company.

Vranas, P. B. M. (2004). Hempel's Raven Paradox: A lacuna in the standard Bayesian solution. *British Journal for the Philosophy of Science*, *55*, 545-560.

Cite this article as:

Saba, W. S. (2020). Language and its Commonsense: Where Formal Semantics Went Wrong, and Where it Can (and Should) Go. *Journal of Knowledge Structures & Systems*, *1*(1), 40-62.

EDITORIAL INFORMATION

Invited article
Editor-in-chief: Luis M. Augusto
Commentators:
Luis M. Augusto
Bill DeSmedt – *Amber Productions, Inc.*
Romeo Issa – *Microsoft*