Concepts, Definitions, and Inheritance

Interpreting the atoms of lexical decomposition

by

DAVID ZORNEK

Applying the Conceptual Interpretation of Atoms to the Generative Lexicon

Thus far, we have approached the linguistic issue, i.e. the problem of interpreting the atoms of lexical decomposition, at a very high theoretical level. An abstract mathematical system has been offered to facilitate the interpretation, complete with a theorem that establishes a way in which semantic atoms and concepts can be regarded as identical. Conversely, we have approached the cognitive science issue from a mostly empirical perspective; the mathematical tools introduced here are offered as a model of empirical results about conceptual hierarchy and typicality. Little has been done to *apply* the mathematics to theories of lexical decomposition, in order to gain understanding of linguistic phenomena, and even less has been done to make use of the formalism as a set of theoretical tools to answer questions for cognitive scientists. In this chapter, we will fill in some remaining gaps and demonstrate, via case study, how to *use* inheritance networks to answer real questions. James Pustejovsky's *Generative Lexicon* [1] is the case we will study.

In **??**, I have already explained the basics of the Generative Lexicon. In the present chapter, I will expand on the earlier explanation by looking more closely at two of Pustejovsky's generative rules. Where linguistics is concerned, Pustejovsky provides a formal model of how word senses are generated according to systematic polysemy, but the model is just that—purely formal. Equipped with the theorem of Chapter 4, we will be able to look "under the hood," so to speak, in order to gain a better understanding of the cognitive mechanism that enables us to understand novel word senses, at least in the cases where word senses are generated according to Pustejovsky's rules.

Cognitive scientists often use linguistic stimuli as a way to study concepts. In some of the experiments outlined in Chapter 3, subjects where primed with sentences and asked to respond. But words (and therefore sentences) are only an intermediary for what cognitive scientists are really interested in, which is concepts. There is very good reason to believe that the same word can stand for different concepts in different contexts (more on this below), and therefore cognitive scientists are faced with the problem of identifying cases where two instances of a word do stand for the same concept, as well as cases where they do not.

The theorem of Chapter 4 is an isomorphism, i.e. a relation-preserving bijection between two domains. These domains, called isomorphs, are regarded as mathematically indistinguishable, and therefore a main value of an isomorphism is that things learned in one domain can be extended into the other. We can answer the linguistic question by looking at how functional roles of concepts vary under generative rules, and we can answer (part of) the cognitive science question by making use of the formalism offered in decompositional lexical semantics. These questions will be answered simultaneously, not due to careful selection of examples, necessarily, but rather because it is a consequence of the theorem that they are in fact the same question approached from different angles.

1 "Josh began the novel."

Let's begin with an example. Consider the sentence "Josh began the novel." The kinds of things we begin are actions; beginning is always beginning *to do* something. But the object of "began," in this sentence, is "the novel," which is not an action. Nevertheless, any native speaker of English will find this to be a perfectly comprehensible sentence. We are, therefore, faced with something of a puzzle: our understanding of the concepts represented by the words in this sentence implies that the sentence should be conceptually incoherent, but the sentence is obviously coherent.

We take the sentence to mean is that Josh is beginning to do something to/with/etc. the novel. There is an implicit action that is being begun, which is not syntactically realized, and "the novel" is the direct object of the verb that refers to this action. For example, two very natural ways of understanding the sentence are "Josh began reading the novel" and "Josh began writing the novel." The conceptual question is whether the concept **novel** is, in fact, an action concept—contrary to intuitions—or whether "the novel" stands for a more complex concept of **reading the novel** or **writing the novel** in this sentence. The linguistic question is how the semantics of "the novel" can allow it to be coherently used as an argument for "began." James Pustejovsky has answered the linguistic question for us, by making use of inheritance networks to provide a formal model of lexical decomposition. By trading on the mathematical indistinguishability of the aspects of semantic content that are represented in inheritance networks and conceptual hierarchy, we can make use of his formal answer to the linguistic question in answering the conceptual question.

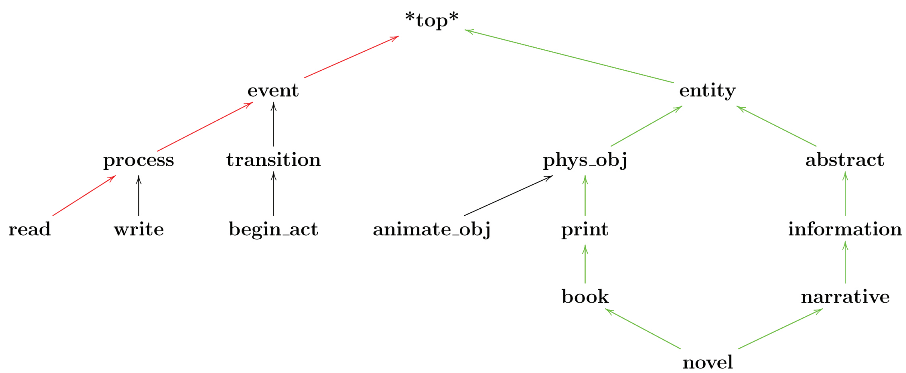
1.1 True Complement Coercion

Recall the following (partial) typed feature structures for "novel" and "begin" from Chapter 2:

$$\begin{bmatrix} \textbf{novel} & \\ \mathcal{A} = & \begin{bmatrix} \text{ARG}_1 = \textbf{x:book} \end{bmatrix} \\ \mathcal{Q} = & \begin{bmatrix} \text{CONST} = \textbf{narrative} \\ \text{FORMAL} = \textbf{book} \\ \text{TELIC} = \textbf{read} \\ \text{AGENT} = \textbf{write} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \textbf{begin} & \\ \mathcal{A} = & \begin{bmatrix} \text{ARG}_1 = \textbf{animate\_obj} \\ \text{ARG}_2 = \textbf{event}_1 \end{bmatrix} \\ \mathcal{E} = & \begin{bmatrix} \text{E}_1 = \textbf{transition} \\ \dots \end{bmatrix} \\ \mathcal{Q} = & \begin{bmatrix} \text{FORMAL} = \textbf{P(event}_1\textbf{,x)} \\ \text{AGENT} = \textbf{begin\_act(event}_1\textbf{,x)} \end{bmatrix} \end{bmatrix}$$

The types given in the typed feature structures are supplied by the following inheritance network (ignore the colors for the time being; we will make use of them shortly):



We can read the argument expectations for "begin" directly off of the typed feature structure. In particular, we see that "begin" expects a direct object of type **event**; this type assignment is a representation of our intuition that events (or, alternatively, actions—we can name the type in any reasonable way that we choose) are the kinds of things that can be begun. Since "novel" is not a verb, as has been explained, it's argument structure specifies the default behavior of "novel" when it is used as an argument; it behaves as an argument of type **book**. But, as can be read off of the inheritance network, **book** does not inherit from **event**, to the sentence "Josh began the novel" entails a violation of the type expectations for arguments of "begin."

According to Pustejovsky, we resolve the type error by applying a generative rule that searches the qualia structure of "begin" for some type that inherits from **event**.[1] In fact, we find that the qualia structure of "novel" includes two qualia, i.e. TELIC and AGENT, whose assigned types inherit from event; we read off of the graph that both **read** and **write** inherit from **event**. Coercing "novel" to behave as an argument of type **read** or **write**

[1]Pustejovsky does offer a formal definition of this rule. His formalization is quite opaque, however, and I think unpacking it will take us needlessly afield from the main point, since the general idea behind the rule can be understood to the degree required here without the formalization.

leads us to understand "Josh began the novel" as "Josh began *to read* the novel" or "Josh began *to write* the novel," respectively.

We might recall here that identity betweeen functional roles is a necessary condition for identity between concepts. The functional role $i(\mathbf{novel})$ is highlighted in green in the above graph. A consistent definition $D_{\mathbf{read}}$ is highlight in red. We can read directly off of the graph that $D_{\mathbf{read}} \notin i(\mathbf{novel})$. But, by the definition of $i(\mathbf{read})$, we know that $D_{\mathbf{read}} \in i(\mathbf{read})$. Therefore, $i(\mathbf{read}) \neq i(\mathbf{novel})$, and therefore the concept being represented by "novel" in "Josh begain the novel" cannot be the concept **novel**. We have answered the conceptual question posed above. I will simply state that this method can be used as a diagnoistic by cognitive scientists who wish to use sentences such as "Josh began the novel" to study the behavior of the concept **novel** (and this method will apply more generally to any context that admits of the above analysis). Cognitive scientists should like to know wether two instances of the linguistic stimulus "novel" cause subjects to make use of some concept other than **novel**.

Note that this kind of diagnostic can only guarantee a non-identity of concepts. Since we lack a sufficient condition for identity of concepts, the above analysis will never establish that two instances of a word actually stand for the same concept. The next example presents a case where no conclusion can be drawn about identity of concepts.

2 "Mary drives a Honda."

The coercion involved here—called subtype coercion—is somewhat trivial, since it involves no type error. Nevertheless, there is still a conceptual question to be answered here. We use "Honda" in two senses: one sense of the word is a proper noun referring to the organization, make of car, etc.; the other sense is a class noun that refers to a subcategory of the category of all cars. We might wonder whether these two senses are actually making use of the same concept, or whether they are a case of the same word being used to represent two distinct concepts. If we can find a sense of "Honda" whose functional role contains a consistent definition $D_{\mathbf{Honda,i}} \notin i(\mathbf{car})$, then we will have identified a sense of "Honda" that represents a concept **Honda′** such that $i(\mathbf{Honda'})\neg \subseteq i(\mathbf{car})$, which would have to mean that we have found a sense of "Honda" that makes use of some concept other than the concept of some subcategory of cars. Let's see whether the typed feature structures Pustejovsky offers us for "drive," "car," and "honda" enable us to draw such a conclusion.

$$\begin{bmatrix} \mathbf{drive} & \\ \mathcal{A} = & \begin{bmatrix} \text{ARG}_1 = \mathbf{x} : \mathbf{human} \\ \text{ARG}_2 = \mathbf{y} : \mathbf{vehicle} \end{bmatrix} \\ \mathcal{E} = & \begin{bmatrix} \text{E}_1 = \mathbf{e}_1 : \mathbf{process} \\ \text{E}_2 = \mathbf{e}_2 : \mathbf{process} \\ \text{RESTR} =< \circ_\propto \end{bmatrix} \\ \mathcal{Q} = & \begin{bmatrix} \text{FORMAL} = \mathbf{move}(\mathbf{e}_2, \mathbf{y}) \\ \text{AGENT} = \mathbf{drive\_act}(\mathbf{e}_1, \mathbf{x}, \mathbf{y}) \end{bmatrix} \end{bmatrix}$$

We can see that "drive" expects a direct object of type **vehicle**. Therefore, the argument-type of "Honda," as it is being used here, must be coercable to **vehicle**. Given the following typed feature structure for "car":

$$\begin{bmatrix} \mathbf{car} & \\ \mathcal{A} = & \begin{bmatrix} \text{ARG}_1 = \mathbf{x} : \mathbf{vehicle} \end{bmatrix} \\ \mathcal{Q} = & \begin{bmatrix} \text{FORMAL} = \mathbf{x} \\ \text{TELIC} = \mathbf{drive}(\mathbf{e}, \mathbf{x}, \mathbf{y}) \\ \text{AGENT} = \mathbf{create}(\mathbf{e}, \mathbf{z}, \mathbf{x}) \end{bmatrix} \end{bmatrix}$$

we see that the argument-type of "car" is **vehicle**, i.e. $\mathbf{car} \sqsubseteq \mathbf{vehicle}$, so the sense of "Honda" being used here is the one that inherits from **car**:

$$\begin{bmatrix} \textbf{Honda} & \\ \mathcal{A} = & \begin{bmatrix} \text{ARG}_1 = \mathbf{x} : \mathbf{car} \end{bmatrix} \\ \mathcal{Q} = & \begin{bmatrix} \text{FORMAL} = \mathbf{x} \\ \text{TELIC} = \mathbf{drive}(\mathbf{e}, \mathbf{x}, \mathbf{y}) \\ \text{AGENT} = \mathbf{create}(\mathbf{e}, \textbf{Honda-Co}, \mathbf{x}) \end{bmatrix} \end{bmatrix}$$

So, we have $\mathbf{Honda} \sqsubseteq \mathbf{car} \sqsubseteq \mathbf{vehicle} \sqsubseteq \ldots$, which means that the consistent definition for **Honda** implicated in "Mary drives a Honda" is $D_{\mathbf{Honda,1}} = \{\mathbf{Honda}, \mathbf{car}, \mathbf{vehicle}, \ldots\}$. If $\mathbf{car} \in D_{\mathbf{Honda,1}}$, then $D_{\mathbf{Honda,1}} \in i(\mathbf{car})$, and we have not identified a sense of "Honda" that makes use of a concept $\mathbf{Honda}'$.

Note that our failure here to establish a sense of "Honda" that is not a subcategory of **car** does not mean that all **Honda** concepts are subcategories of **car**, or that all **Honda** concepts are the same. Nor have we even established that *this* **Honda** concept is a subcategory of **car**. The diagnostic used here is inconclusive. Although $D_{\mathbf{Honda,1}} \in i(car)$, there may be other $D_{\mathbf{Honda}}, i) \in i(\mathbf{Honda})$ that are not members of $i(\mathbf{car})$. If one of these does exist, then the **Honda** concept being used here is not a subcategory of **car**. However, establishing this would require having identified a sufficient condition for identity of concepts, which we currently lack.

3 Summary

We can make use of the isomorphism to study semantic atoms by looking at concepts and vice versa. Taking GL as a case study allows us to see how the functional role i can provide us with diagnostics to determine whether two distinct uses of a word represent the same concept. In making use of diagnostics to establish that the functional roles of the concepts being represented by words are not identical, we can infer that the concepts are not identical. However, the framework of inheritance networks affords us with no known diagnostic to establish that two concepts *are* identical. Such a diagnostic would require possession of a sufficient condition for identity of concepts, but the tools defined here can only provide us with a necessary condition for identity of concepts.

References

[1] James Pustejovsky. *The Generative Lexicon.* The MIT Press, 1998.