Feature Logic for Dotted Types: A Formalism for Complex Word Meanings

Manfred Pinkal and Michael Kohlhase
Universität des Saarlandes, Germany
{kohlhase@ags|pinkal@coli}.uni-sb.de

Abstract

In this paper we revisit Pustejovsky's proposal to treat ontologically complex word meaning by so-called dotted pairs. We use a higher-order feature logic based on Ohori's record λ-calculus to model the semantics of words like *book* and *library*, in particular their behavior in the context of quantification and cardinality statements.

1 Introduction

The treatment of lexical ambiguity is one of the main problems in lexical semantics and in the modeling of natural language understanding. Pustejovsky's framework of the "Generative Lexicon" made a contribution to the discussion by employing the concept of type coercion, thus replacing the enumeration of readings by the systematic context-dependent generation of suitable interpretations, in the case of systematic polysemies (Pustejovsky, 1991; Pustejovsky, 1995). Also, Pustejovsky pointed to a frequent and important phenomenon in lexical semantics, which at first sight looks as another case of polysemy, but is significantly different in nature.

(1) The book is blue/on the shelf.

(2) Mary burned the book.

(3) The book is amusing.

(4) Mary understands the book.

(5) The book is beautiful.

(6) Mary likes the book.

(7) Mary read the book.

Examples (1)-(4) suggest an inherent ambiguity of the common noun *book*: *blue*, *on the shelf*, and *burn* subcategorize for a physical object, while *amusing* and *understand* require an informational object as argument. (5) and (6) are in fact ambiguous: The statements may refer either to the shape or the content of the book. However, a thorough analysis of the situation shows that there is a third reading where the beauty of the book as well as Mary's positive attitude are due to the harmony between physical shape and informational content. The action of reading, finally, is not carried out on a physical object alone, nor on a pure informational object as argument, but requires an object which is essentially a combination of the two. This indicates a semantic relation which is conjunctive or additive in character, rather than a disjunction between readings as in the ambiguity case. In addition to the more philosophical argument, the assumption of a basically different semantic relation is supported by observations from semantic composition. If the physical/informational distinction in the semantics of *book* were just an ambiguity, (8) and (9) would not be consistently interpretable, since the sortal requirements of the noun modifier (*amusing* and *on the shelf*, resp.) are incompatible with the selection restrictions of the verbs *burn* and *understand*, respectively.

(8) Mary burned an amusing book.

(9) Mary understands the book on the shelf.

Pustejovsky concludes that ontologically complex objects must be taken into account to describe lexical semantics properly, and he represents them as "dotted pairs" made up form two (or more) ontologically simple objects, and being semantically categorized as "dotted types", e.g., $\mathbb{P} \bullet \mathbb{I}$ in the case of *book*. He convincingly argues that complex types are omnipresent in the lexicon, the physical/informational object distinction being just a special case of a wide range of dotted types, including container/content (*bottle*), aperture/panel (*door*) building/institution (*library*).

The part of the Generative Lexicon concept which was *not* concerned with ontologically complex objects, i.e., type coercion and co-composition mechanisms using so-called qualia information, has triggered a line of intensive and fruitful research in lexical semantics, which led to progress in representation formalisms and tools for the computational lexicon (see e.g. (Copestake and Briscoe, 1995; Dölling, 1995; Busa and Bouillon, forthcoming; Egg, 1999)). In contrast, a problem with Pustejovsky's proposal about the complex objects is that the dotted-pair notation has been formally and semantically not clear enough to form a starting point for meaning representation and processing.

In this paper, we present a formally sound semantic reconstruction of complex objects, using a higher-order feature logic based on Ohori's record λ-calculus (1995) which has been originally developed for functional- and object-oriented programming. We do not claim that our reconstruction provides a full theory of the of the peculiar kind of ontological objects, but it appears to be useful as a basis for representing lexical entries for these objects and modeling the composition process in which they are involved. We will not only show that the basic examples above can be treated, but also that our treatment provides a straightforward solution to some puzzles concerning the behavior of dotted pairs in quantificational, cardinality and identity statements.

(10) Mary burned every book in the library.

(11) Mary understood every book in the library.

(12) There are 2000 books in the library.

(13) All new books are on the shelf.

(14) The book on your book-shelf is the one I saw in the library.

In (10), the quantification is about physical objects, whereas in (11), it concerns the books qua informational unit. (12) is ambiguous between a number-of-copies and a number-of-titles reading. The respective readings in (10) and (11) appear to be triggered by the sortal requirements of the verbal predicate, as the ambiguity in (12) is due to the lack of a selection restriction. However, (13) – uttered towards a customer in a book store – has a natural reading where the quantification relates to the information level and the predicate is about physical objects. Finally, (14) has a reading where a relation of non-physical identity is ascribed to objects which are both referred to by physical properties.

2 The Record-λ-Calculus $\mathcal{F}^{\leq}$

In order to reduce the complexity of the calculus, we will first introduce a feature λ-calculus $\mathcal{F}$ and then extend it to $\mathcal{F}^{\leq}$. $\mathcal{F}$, is an extension of the simply typed λ-calculus by feature structures (which we will call records). See Figure 1 for the syntactical categories of the raw terms.

We assume the base types e (for individuals) and t (for truth values), and a set $\mathcal{L} = \{\ell_1, \ell_2, \ldots\}$ of features. The set of well-typed

$$
\begin{array}{lll}
\mathcal{T} & ::= & e \mid t \mid \mathcal{T} \to \mathcal{T}' \\
 & & \mid \{\!\{\ell^1 : \mathcal{T}^1, \ldots, \ell^n : \mathcal{T}^n\}\!\} \\
 & & \text{(Types: } \alpha, \beta, \ldots) \\
\mathbf{M} & ::= & X \mid c \mid (\mathbf{MN}) \mid \lambda X_{\mathcal{T}}.\mathbf{M} \mid \mathbf{M}.\ell \\
 & & \mid \{\!\{\ell^1 = \mathbf{M}_1, \ldots, \ell^n = \mathbf{M}_n\}\!\} \\
 & & \text{(Formulae } \mathbf{A}, \mathbf{B}, \ldots) \\
\Sigma & ::= & \emptyset \mid \Sigma, [c : \mathcal{T}] \quad \text{(Signature)} \\
\Gamma & ::= & \emptyset \mid \Gamma, [X : \mathcal{T}] \quad \text{(Environment)}
\end{array}
$$

Figure 1: Syntax

terms is defined by the inference rules in Figure 2 for the typing judgment $\Gamma \vdash_\Sigma \mathbf{A} : \alpha$. The meaning of this judgment is that term $\mathbf{A}$ has type $\alpha \in \mathcal{T}$ relative to the (global) type assumptions in the **signature** Σ and the (local) type assumptions Γ (the **context**) for the variables. As usual, we say that a term $\mathbf{A}$ is of type α (and often simply write $\mathbf{A}_\alpha$ to indicate this), iff $\Gamma \vdash_\Sigma \mathbf{A}\colon \alpha$ is derivable by these rules. We will call a type a **record type** (with **features** ℓ_i), iff it is of the form $\{\!\{\ell_1\colon \alpha_1, \ldots, \ell_n\colon \alpha_n\}\!\}$. Similarly, we call an $\mathcal{F}$-term $\mathbf{A}$ a **record**, iff it has a record type. Note that record **selection** operator "." can only be applied to records. In a slight abuse of notation, we will also use it on record types and have $\mathbf{A}_\alpha.\ell\colon \alpha.\ell$.

$$\frac{[c\colon \alpha] \in \Sigma}{\Gamma \vdash_\Sigma c\colon \alpha} \qquad \frac{[X\colon \alpha] \in \Gamma}{\Gamma \vdash_\Sigma X\colon \alpha}$$

$$\frac{\Gamma \vdash_\Sigma \mathbf{A}\colon \gamma \to \alpha \quad \Gamma \vdash_\Sigma \mathbf{C}\colon \gamma}{\Gamma \vdash_\Sigma \mathbf{AC}\colon \alpha}$$

$$\frac{\Gamma, [X\colon \beta] \vdash_\Sigma \mathbf{A}\colon \alpha}{\Gamma \vdash_\Sigma \lambda X_\beta.\mathbf{A}\colon \beta \to \alpha}$$

$$\frac{\Gamma \vdash_\Sigma \mathbf{A}\colon \{\!\{\ldots, \ell\colon \alpha, \ldots\}\!\}}{\Gamma \vdash_\Sigma \mathbf{A}.\ell\colon \alpha}$$

$$\frac{\Gamma \vdash_\Sigma \mathbf{A}_1\colon \alpha_1 \quad \ldots \quad \Gamma \vdash_\Sigma \mathbf{A}_n\colon \alpha_n}{\Gamma \vdash_\Sigma \{\!\{\ell_1 = \mathbf{A}_1, \ldots, \ell_n = \mathbf{A}_n\}\!\}}$$

Figure 2: Well-typed terms in $\mathcal{F}$

It is well-known that type inference with these rules is decidable (as a consequence we will sometimes refrain from explicitly marking types in our examples), that well-typed terms have unique types, and that the calculus admits subject reduction, i.e that the set of well-typed terms is closed under well-typed substitutions.

The calculus $\mathcal{F}$ is equipped with an (operational) equality theory, given by the rules in Figure 3 (extended to congruence relations on $\mathcal{F}$-terms in the usual way). The first two are just the well-known $\beta\eta$ equality rules from λ-calculus (we assume alphabetic renaming of bound variables wherever necessary). The second two rules specify the semantics of the record dereferencing operation ".". Here we know that these rules form a canonical (i.e. terminating and confluent), and type-safe (reduction does not change the type) reduction system, and that we therefore have unique $\beta\eta\rho$-normal forms. The semantics of $\mathcal{F}^{\leq}$ is a straightforward extention of that of the simply typed λ-calculus: records are interpreted as partial functions from features to objects, and dereferencing is only application of these functions. With this semantics it is easy to show that the evaluation mapping is well-typed ($\mathcal{I}_\varphi(\mathbf{A}_\alpha) \in \mathcal{D}_\alpha$) and that the equalities in Figure 3 are sound (i.e. if $\mathbf{A} =_{\beta\eta\rho} \mathbf{B}$, then $\mathcal{I}_\varphi(\mathbf{A}) = \mathcal{I}_\varphi(\mathbf{B})$).

$$\frac{}{(\lambda X_\beta.\mathbf{A})\mathbf{B} \longrightarrow_\beta [\mathbf{B}/X]\mathbf{A}}$$

$$\frac{X \notin \mathbf{free}(\mathbf{A})}{(\lambda X.\mathbf{A}X) \longrightarrow_\eta \mathbf{A}}$$

$$\frac{}{\{\!\{\ldots, \ell = \mathbf{A}, \ldots\}\!\}.\ell \to_\rho \mathbf{A}}$$

$$\frac{\Gamma \vdash_\Sigma \mathbf{A}\colon \{\!\{\ell_1\colon \alpha_1, \ldots, \ell_n\colon \alpha_n\}\!\}}{\{\!\{\ell_1 = \mathbf{A}.\ell_1, \ldots, \ell_n = \mathbf{A}.\ell_n\}\!\} \to_\eta \mathbf{A}}$$

Figure 3: Operational Equality for $\mathcal{F}$.

Up to now, we have a calculus for so-called **closed records** that exactly *prescribe* the features of a record. The semantics given above also licenses a slightly different interpretation: a record type $\rho = \{\!\{\ell_1\colon \alpha_n, \ldots, \ell_n\colon \alpha_n\}\!\}$ is *descriptive*, i.e. an $\mathcal{F}$-term of type ρ would only be required to have *at least* the features $\ell_1, \ldots \ell_n$, but may actually have more. This makes it necessary to introduce a **subtyping relation** $\leq$, since a record $\{\!\{\ell = \mathbf{A}_\alpha\}\!\}$ will now have the types $\{\!\{\ell\colon \alpha\}\!\}$ and $\{\!\{\}\!\}$. Of course we have $\{\!\{\ell\colon \alpha\}\!\} \leq \{\!\{\}\!\}$, since the latter is less restrictive. The higher-order feature logic $\mathcal{F}^{\leq}$ we will use for the linguistic analysis in section 3 is given as $\mathcal{F}$ extended by the rules in Figure 4. The first rule specifies that record

$$\frac{k \leq n}{\{\!\{\ell_1 : \alpha_1, \ldots, \ell_n : \alpha_n\}\!\} \leq \{\!\{\ell_1 : \alpha_1, \ldots, \ell_n : \alpha_k\}\!\}}$$

$$\frac{\Gamma \vdash_\Sigma \mathbf{A} : \alpha \quad \alpha \leq \beta}{\Gamma \vdash_\Sigma \mathbf{A} : \beta}$$

$$\frac{\alpha \in \mathcal{BT}}{\alpha \leq \alpha} \qquad \frac{\alpha \leq \alpha' \quad \beta \leq \beta'}{\alpha' \to \beta \leq \alpha \to \beta'}$$

Figure 4: The open record calculus $\mathcal{F}^{\leq}$

types that prescribe more features are more specific, and thus describe a smaller set of objects. The second rule is a standard weakening rule for the subtype relation. We need the reflexivity rule for base types in order to keep the last rule, which induces the subtype relation on function types from that of its domain and range types simple. It states that function spaces can be enlarged by enlarging the range type or by making the domain smaller (intuitively, every function can be restricted to a smaller domain). We say that $\leq$ is **covariant** (preserving the direction) in the range and **contravariant** in the domain type (inverting the direction).

For $\mathcal{F}^{\leq}$, we have the same meta-logical results as for $\mathcal{F}^{\leq}$ (the type-preservations, subject reduction, normal forms, soundness,...) except for the unique type property, which cannot hold by construction. Instead we have the **principal type** property, i.e. every $\mathcal{F}^{\leq}$-term has a unique minimal type.

To fortify our intuition about $\mathcal{F}^{\leq}$, let us take a look at the following example: It should be possible to apply a function $\mathbf{F}$ of type $\{\!\{\ell_1 : \alpha\}\!\} \to \beta$ to a record with features ℓ_1, ℓ_2, since $\mathbf{F}$ only expects ℓ_1. The type derivation in Figure 5 shows that $\mathbf{F}\{\!\{\ell_1 = \mathbf{A}^1_{\alpha_1}, \ell_2 = \mathbf{A}^2_{\alpha_2}\}\!\}$ is indeed well-typed. In the first block, we use the rules from Figure 4 (in particular contravariance) to establish a subtype relation that is used in the second block to weaken the type of $\mathbf{F}$, so that it (in the third block) can be applied to the argument record that has one feature more than the feature ℓ_1 required by $\mathbf{F}$'s type.

$$\cfrac{\cfrac{1 \leq 2}{\{\!\{\ell_1 : \alpha_1, \ell_2 : \alpha_2\}\!\} \leq \{\!\{\ell_1 : \alpha_1\}\!\}}}{\{\!\{\ell_1 : \alpha_1\}\!\} \to \beta \leq \{\!\{\ell_1 : \alpha_1, \ell_2 : \alpha_2\}\!\} \to \beta}$$

$$\frac{\mathbf{F} : \{\!\{\ell_1 : \alpha_1\}\!\} \to \beta \quad *}{\mathbf{F} : \{\!\{\ell_1 : \alpha_1, \ell_2 : \alpha_2\}\!\} \to \beta}$$

$$\cfrac{* \quad \cfrac{\Gamma \vdash_\Sigma \mathbf{A}^i : \alpha_i}{\Gamma \vdash_\Sigma \{\!\{\ell_1 = \mathbf{A}^1_{\alpha_1}, \ell_2 = \mathbf{A}^2_{\alpha_2}\}\!\} : \{\!\{\ell_1 : \alpha_1, \ell_2 : \alpha_2\}\!\}}}{\Gamma \vdash_\Sigma \mathbf{F}\{\!\{\ell_1 = \mathbf{A}^1_{\alpha_1}, \ell_2 = \mathbf{A}^2_{\alpha_2}\}\!\} : \beta}$$

Figure 5: A $\mathcal{F}^{\leq}$ example derivation

3 Modeling ontologically complex objects

We start with the standard Montagovian analysis (Montague, 1974), only that we base it on $\mathcal{F}^{\leq}$ instead of the simply typed λ-calculus.

For our example, it will be sufficient to take the set $\mathcal{L}$ of features as a superset of $\{\mathbb{P}, \mathbb{I}, \mathbb{H}\}$ (where the first stand for physical, and informational facets of an object). In our fragment we use the extension $\mathcal{F}^{\leq}$ to structure type e into subsets given by types of the form $\{\!\{\ell_1 : e, \ldots, \ell_n : e\}\!\}$. Note that throwing away all feature information and mapping each such type to a type E in our examples will yield a standard Montagovian treatment of NL expressions, where E takes the role that e has in standard Montague grammar. Linguistic examples are the proper name *Mary*, which translates to $\mathsf{mary}' : \{\!\{\mathbb{H} : e\}\!\}$, *shelf* which translates to $\mathsf{shelf}' : \{\!\{\mathbb{P} : e\}\!\} \to t$, and the common noun *book* which translates to $\mathsf{book}' : \{\!\{\mathbb{P} : e, \mathbb{I} : e\}\!\} \to t$.

A predicate like *blue* requires a physical object as argument. To be precise, the argument need not be an object of type $\{\!\{\mathbb{P} : e\}\!\}$, like a shelf or a table. *blue* can be perfectly applied to complex objects as books, libraries, and doors, if they have a physical realization, irrespective of whether it is accompanied by an informational object, an institution, or an aperture. At first glance, this seems to be a significant difference from kind predicates like *shelf* and *book*. However, it is OK to interpret the type assignment for kind predicates along with property denoting expressions: In both cases, the occurrence of a feature ℓ means that ℓ occurs in the type of the argument object. Thus, $\{\!\{\ell : e\}\!\} \rightarrow t$ is a sortal characterization for a predicate A with the following impact:

1. A has a value for feature ℓ, possibly among other features,
2. the semantics of A is projective, i.e., the applicability conditions of A and accordingly the truth value of the resulting predication is only dependent of the value of ℓ.

Note that 1. is exactly the behavior that we have built the extension $\mathcal{F}^{\leq}$ for and that we have discussed with the example in Figure 5. We will now come to 2.

Although type e never occurs as argument type directly in the translation of NL expressions, representation language constants with type-e arguments are useful in the definition of the semantics of lexical entries. E.g., the semantics of *book* can be defined using the basic constant $book^*$ of type $e \rightarrow e \rightarrow t$, as $\lambda x.(book^*(x.\mathbb{P}, x.\mathbb{I}))$, where $book^*$ expresses the book-specific relation holding between physical and informational objects[1].

The fragment in Figure 6 provides representations for some of the lexical items occurring in the examples of Section 1, in terms of the basic expressions

$$mary^*{:}\, e, \quad shelf^*, blue^*, amusing^*{:}\, e \rightarrow t$$
$$on^*, book^*, burn^*, understand^*{:}\, e \rightarrow e \rightarrow t,$$
$$read^*{:}\, e \rightarrow e \rightarrow e \rightarrow t$$

Observe that the representations nicely reflect the distinction between linguistic arity of the lexical items, which is given by the λ-prefix (e.g., two-place in the case of *read*), and

Word	Meaning/Type
Mary	$\{\!\{\mathbb{H} = mary^*\}\!\}{:}\, \{\!\{\mathbb{H}{:}\, e\}\!\}$
shelf	$\lambda x.(shelf^*(x.\mathbb{P})){:}\, \{\!\{\mathbb{P}{:} e\}\!\} \rightarrow t$
book	$\lambda x.book^*(x.\mathbb{P}, x.\mathbb{I})$ $\{\!\{\mathbb{P}{:} e, \mathbb{I}{:} e\}\!\} \rightarrow t$
amusing	$\lambda x.amusing^*(x.\mathbb{I})$ $\{\!\{\mathbb{I}{:} e\}\!\} \rightarrow t$
on	$\lambda xy.on^*(x.\mathbb{P}, y.\mathbb{P})$ $\{\!\{\mathbb{P}{:} e\}\!\} \rightarrow \{\!\{\mathbb{P}{:} e\}\!\} \rightarrow t$
burn	$\lambda xy.burn^*(x.\mathbb{H}, y.\mathbb{P})$ $\{\!\{\mathbb{P}{:} e\}\!\} \rightarrow \{\!\{\mathbb{P}{:} e\}\!\} \rightarrow t$
underst.	$\lambda xy.understand^*(x.\mathbb{H}, x.\mathbb{I})$ $\{\!\{\mathbb{H}{:}\, e\}\!\} \rightarrow \{\!\{\mathbb{I}{:} e\}\!\} \rightarrow t$
read	$\lambda xy.read^*(x.\mathbb{H}, y.\mathbb{P}, y.\mathbb{I})$ $\{\!\{\mathbb{H}{:}\, e\}\!\} \rightarrow \{\!\{\mathbb{P}{:} e, \mathbb{I}{:} e\}\!\} \rightarrow t$

Figure 6: A tiny fragment of English

the "ontological arity" of the underlying basic relations (e.g., the 3-place-relation holding between a person, the physical object which is visually scanned, and the content which is acquired by that action). In particular, all of the meanings are projective, i.e. they only pick out the features from the complex arguments and make them available to the basic predicate. Therefore, we can reconstruct the meaning term $\mathbf{R} = \lambda xy.read^*(x.\mathbb{H}, y.\mathbb{P}, y.\mathbb{I})$ of *read* if we only know the relevant features (we call them selection restrictions) of the arguments, and write $\mathbf{R}$ as $read^*[\{\mathbb{H}\}\{\mathbb{P}, \mathbb{I}\}]$.

The interpretation of sentence (2) via basic predicates is shown in (15) to (17). For simplicity, the definite noun phrase is translated by an existential quantifier here. (15) shows the result of the direct one-to-one-translation of lexical items into representation language constants. In (16), these constants are replaced by λ-terms taken from the fragment. (17) is obtained by β-reduction and η-equality from (16): in particular, $\{\!\{\mathbb{H} = mary^*.\mathbb{H}\}\!\}$ is replaced by the ρ-equivalent $mary^*$.

(15) $\exists v.book'(v) \wedge burn'(\{\!\{\mathbb{H} = mary^*\}\!\}, v)$

(16) $\exists v.(\lambda x.book^*(x.\mathbb{P}, x.\mathbb{I}))(v) \wedge$
$(\lambda xy.burn^*(x.\mathbb{H}, x.\mathbb{P}))(\{\!\{\mathbb{H} = mary^*\}\!\}, v)$

(17) $\exists v.book^*(v.\mathbb{P}, v.\mathbb{I}) \wedge burn^*(mary^*, v.\mathbb{P})$

[1] Pustejovsky conjectures that the relation holding among different ontological levels is more than just a set of pairs. We restrict ourselves to the extensional level here.

(18) and (19) as semantic representations for (4) and (7), respectively, demonstrate how the predicates *understand* and *read* pick out objects of appropriate ontological levels. (20) and (21) are interpretations of (8) and (9) respectively, where nested functors coming with different sortal constraints apply to one argument. The representations show that the functors select there appropriate ontological level locally, thereby avoiding global inconsistency.

(18) $\exists v(book^*(v.\mathbb{P}, v.\mathbb{I})) \wedge$
$(understand^*(mary^*, v.\mathbb{I}))$

(19) $\exists v(book^*(v.\mathbb{P}, v.\mathbb{I})) \wedge$
$(read^*(mary^*, v.\mathbb{P}, v.\mathbb{I}))$

(20) $\exists v(book^*(v.\mathbb{P}, v.\mathbb{I})) \wedge amusing^*(v.\mathbb{I}) \wedge$
$(burn^*(mary^*, v.\mathbb{P}))$

(21) $\exists v(book^*(v.\mathbb{P}, v.\mathbb{I})) \wedge \exists u shelf^*(v.\mathbb{P}) \wedge$
$on^*(v.\mathbb{P}, u.\mathbb{P}) \wedge$
$(understand^*(mary^*, v.\mathbb{I}))$

The lexical items *beautiful* and *like* in (5) and (6), resp., are polysemous because of the lack of strict sortal requirements. They can be represented as relational expressions containing a parameter for the selection restrictions which has to be instantiated to a set of features by context. *like*, e.g., can be translated to $like[\mathcal{S}]'$, with $like[\{\mathbb{P}\}]'$, $like[\{\mathbb{I}\}]'$, and $like[\{\mathbb{P}, \mathbb{I}\}]'$ as (some of the) possible readings. Of course this presupposes the availability of a set of basic predicates $like_i^*$ of different ontological arities.

4 Quantifiers and Cardinalities

We now turn to the behavior of non-existential quantifiers and cardinality operators in combination with complex objects. The choice of the appropriate ontological level for an application of these operators may be guided by the sortal requirements of the predicates used (as in (10)-(12)), but as (13) demonstrates it is not determined by the lexical semantics. We represent quantifiers and cardinality operators as second-order relations, according to the theory of generalized quantifiers (Montague, 1974; Barwise and Cooper, 1981) and take them to be parameterized by a context variable $\mathcal{S} \subseteq \mathcal{L}$ for selection restrictions in the same manner as the predicates *like* and *beautiful*. The value of $\mathcal{S}$ may depend on the general context as well as on semantic properties of lexical items in the utterance.

We define the semantics of a parameterized quantifier $\mathcal{Q}|_{\mathcal{S}}$ by applying its respective basic, non-parameterized variants to the $\mathcal{S}$-projections of their argument predicates P and Q to features in $\mathcal{S}$, which we write as $P|_{\mathcal{S}}$ and $Q|_{\mathcal{S}}$, respectively. Formally $P|_{\{\ell_1,\ldots,\ell_n\}}$ is

$$\lambda x_1 \ldots x_n.\exists u.P(u) \wedge x_1 = u.\ell_1 \wedge \ldots \wedge x_n = u.\ell_n$$

A first proposal is given in (22). (23) gives the representation of sentence (13) in the "bookstore reading" (omitting the semantics of *new* and representing *on the shelf* as an atomic one-place predicate, for simplicity), (24) the reduction of (23) to ordinary quantification on the $\mathcal{S}$-projections, which is equivalent to the first-order formula (25), which in turn can be spelled out as (26) using basic predicates.

(22) $\mathcal{Q}|_{\mathcal{S}}(P, Q) \Leftrightarrow \mathcal{Q}\,(P|_{\mathcal{S}}, Q|_{\mathcal{S}})$

(23) $every|_{\{\mathbb{I}\}}(book', on\_shelf')$

(24) $every^*\left(book'|_{\{\mathbb{I}\}}, on\_shelf'|_{\{\mathbb{I}\}}\right)$

(25) $\forall x.\exists u.(x = u.\mathbb{I} \wedge book'(u))$
$\Longrightarrow \exists v.x = v.\mathbb{I} \wedge on\_shelf'(v)$

(26) $\forall x.\exists u.(x = u.\mathbb{I} \wedge book^*(u.\mathbb{P}, u.\mathbb{I}))$
$\Longrightarrow \exists v.x = v.\mathbb{I} \wedge on\_shelf^*(v.\mathbb{P})$

As one can easily see, the instantiation of $\mathcal{S}$ to $\{\mathbb{I}\}$ triggers the wanted $\forall\exists$ reading ("for all books (as informational objects) there is a physical object on the shelf"), where the instantiation to $\{\mathbb{P}\}$ would have given the $\forall\forall$ reading, since on_shelf' is projective for $\mathbb{P}$ only, and as a consequence we have

$$\begin{aligned}
&on_shelf'|_{\{\mathbb{P}\}} \\
&= \lambda x.\exists u.on_shelf'(u) \wedge x = u.\mathbb{P} \\
&= \lambda x.\exists u.on_shelf^*(u.\mathbb{P}) \wedge x = u.\mathbb{P} \\
&\Leftrightarrow \lambda x.\exists u.on_shelf^*(x) \wedge x = u.\mathbb{P} \\
&\Leftrightarrow \lambda x.on_shelf^*(x)
\end{aligned}$$

The extension to cases (10)-(12) is straightforward.

The proposed interpretation may be too permissive. Take a situation, where new publications are alternatively available as book and on CD-ROM. Then (22)-(26) may come out to be true even if no book at all is on the shelf (only one CD-ROM containing all new titles). We therefore slightly modify the general scheme (22) by (27), where the restriction of the quantifier is repeated in the nuclear scope.

(27) $\mathcal{Q}|_{\mathcal{S}}(P, Q) \Leftrightarrow$
$\mathcal{Q}\,(P|_{\mathcal{S}}, (\lambda x.P(x) \wedge B(x))|_{\mathcal{S}})$

For ordinary quantification, this does not cause any change, because of the monotonicity of NL quantifiers. In our case of level-specific quantification, it guarantees that the second argument covers only projections originating from the right type of complex objects. We give the revised first-order representation corresponding to (26) in (28).

(28) $\forall x.\exists u.(x = u.\mathbb{I} \wedge \mathit{book}^*(u.\mathbb{P}, u.\mathbb{I}))$
$\Longrightarrow \exists v.x =$
$v.\mathbb{I} \wedge \mathit{book}^*(v.\mathbb{P}, v.\mathbb{I}) \wedge \mathit{on\_shelf}^*(v.\mathbb{P})$

5 Conclusion

Our higher-order feature logic $\mathcal{F}^{\leq}$ provides a framework for the simple and straightforward modeling of ontologically complex objects, including the puzzles of quantification and cardinality statements. In this framework, a number of interesting empirical questions can be further pursued:

The ontology for complex objects can be investigated. So far, we constrained ourselves to the simplest case of "dotted pairs", and may even have taken over a wrong classification from the literature, talking about the dualism of physical and informational objects, where a type/token distinction might have been more adequate. The reality about books (as well as bottles and libraries) might be more complex, however, including both the $\mathbb{P}/\mathbb{I}$ distinction as well as hierarchical type/token structures.

The linguistic selection restrictions are probably more complex than we assumed in this paper: As Pustejovsky argues (1998), we may have to take distinguish exocentric and endocentric cases of dotted pairs, as well as projective and non-projective verbal predicates.

Another fruitful question might be whether the framework could be used to reconsider the mechanism of type coercion in general: It may be that at least some cases of reinterpretation may be better described by adding an ontological level, and thus creating a complex object, rather than by switching from one level to another.

We would like to conclude with a very general remark: The data type of feature structures as employed in our formalism has been widely used in grammar formalisms, among other things to incorporate semantic information. In this paper, a logical framework for semantics is proposed, which itself has feature structures as a part of the meaning representation. It may be worthwhile to consider whether this property can be used to tell a new story about treating syntax and semantics in a uniform framework.

References

John Barwise and Robin Cooper. 1981. Generalized quantifiers and natural language. *Linguistics and Philosophy*, 4:159–219.

F. Busa and P. Bouillon, editors. forthcoming. *The language of word meaning.* Cambridge University Press, Cambridge.

A. Copestake and T. Briscoe. 1995. Semi-productive polysemy and sense extension. *Journal of Semantics*, 12:15–67.

J. Dölling. 1995. Ontological domains, semantic sorts and systematic ambiguity. *Int. Journal of Human-Computer Studies*, 43:785–807.

Markus Egg. 1999. Reinterpretation from a synchronic and diachronic point of view. Submitted.

R. Montague. 1974. The proper treatment of quantification in ordinary english. In R. Montague, editor, *Formal Philosophy. Selected Papers.* Yale University Press, New Haven.

Atsushi Ohori. 1995. A polymorphic record calculus and its compilation. *ACM Transac-*

tions on Programming Languages and Systems, 17(6):844–895.

James Pustejovsky. 1991. The generative lexicon. *Computational Linguistics*, 17.

James Pustejovsky. 1995. *The Generative Lexicon*. MIT Press, Cambridge, MA.

James Pustejovsky. 1998. The semantics of lexical underspecification. *Folia Linguistica*, 32:323–347.