J. Linguistics **53** (2017), 701–750.
doi:10.1017/S0022226716000189 First published online 5 July 2016

Representing argument structure[1]

LIVNAT HERZIG SHEINFUX

University of Haifa

NURIT MELNIK

The Open University

SHULY WINTNER

University of Haifa

(Received 30 January 2015; revised 16 June 2016)

Existing approaches to the representation of argument structure in grammar tend to focus either on semantics or on syntax. Our goal in this paper is to strike the right balance between the two levels by proposing an analysis that maintains the independence of the syntactic and semantic aspects of argument structure, and, at the same time, captures the interplay between the two levels. Our proposal is set in the context of the development of a large-scale grammar of Modern Hebrew within the framework of Head-driven Phrase Structure Grammar (HPSG). Consequently, an additional challenge it faces is to reconcile two conflicting desiderata: to be both linguistically coherent *and* realistic in terms of the grammar engineering effort. We present a novel representation of argument structure that is fully implemented in HPSG, and demonstrate its many benefits to the coherence of our Hebrew grammar. We also highlight the additional dimensions of linguistic generalization that our proposal provides, which we believe are also applicable to grammars of other languages.

KEYWORDS: argument structure, Hebrew grammar, HPSG, Modern Hebrew, semantic selection, syntactic selection, syntax-semantics interface

1. Introduction

Syntax and semantics play different roles in the relationship between a predicate and its arguments. From a semantic perspective, predicates require different

[1] This research was supported by the Israel Science Foundation (grant no 505/11). We are indebted to Petter Haugereid for his contribution to this project in its earlier stages, and to Tali Arad Greshler and Adam Przepiórkowski for their help and advice with previous drafts of this paper. We are grateful to the *Journal of Linguistics* anonymous reviewers for many helpful and constructive comments. All remaining errors and misconceptions are, of course, our own.

This paper is dedicated to the memory of Chuck Fillmore (1929–2014) and Ivan Sag (1949–2013).

Abbreviations used in this paper for agreement are 1/2/3 = person; S/P = number; F/M = gender. In addition ACC = accusative case.

arguments to assume particular semantic functions or roles. This is referred to as *semantic selection (s-selection)* (Pesetsky 1996). *Syntactic selection (c-selection)* determines the syntactic categories that realize the predicate's arguments. A related notion, *subcategorization*, pertains to the idea that lexemes, which are *categorized* according to syntactic categories, can be further *subcategorized* according to the type and number of syntactic complements that they require. Finally, *linking* describes the matching between arguments and syntactic functions. Thus, for example, active verbs link their Agent argument with the Subject role, while with passive verbs, the Patient argument assumes this role.

The question of whether s-selection, c-selection, and linking are basic or whether one is reducible to another has been worrying linguists since at least Grimshaw (1979), who argues that semantic selection and syntactic selection are two independent sets of restrictions. She focuses on the selection of interrogative and exclamatory complements, and shows that syntactic subcategorization alone cannot account for the data.

As an illustration of her argument consider the following sentences:

(1) (a) John asked me [$_{CP}$ what the time was].
(b) John asked me [$_{NP}$ the time].
(c) John wondered [$_{CP}$ what the time was].
(d) *John wondered [$_{NP}$ the time].

Both *ask* and *wonder* can appear with an embedded question as a complement, yet only *ask* is compatible with an NP complement. In this context, this NP is referred to as a 'concealed question' since it can be intuitively paraphrased as an embedded question.[2] Nevertheless, this set of sentences raises the question of what drives the distinction between the subcategorization options of the two verbs. Grimshaw (1979) claims that these differences cannot be derived from semantic differences, and thus suggests that syntactic and semantic selection are independent.

Predicates, in Grimshaw's system, impose two types of restrictions on their complements. Semantic selection involves semantic categories such as *Q* (question), *P* (proposition), or *E* (exclamation). Syntactic subcategorization targets syntactic phrase types such as CP, PP, NP, AdjP. The two sets of restrictions pertaining to *ask* and *wonder* are demonstrated in Table 1.

	Semantic selection	Syntactic subcategorization
Ask	<Q>	[CP,NP]
Wonder	<Q>	[CP]

Table 1
The independence of semantic and syntactic selection.

Although Grimshaw's claim is that the two restrictions are independent, naturally the association between them is not completely arbitrary. Grimshaw (1979)

[2] Baker (1968) goes as far as proposing that concealed questions are base-generated as a sentence and undergo a process of ellipsis that renders them NPs.

suggests that each semantic category is associated with a *Canonical Structural Realization* (CSR), which is its ‘default’ syntactic realization. Thus, for example, the CSR of *Q* arguments is CP, and the CSR of ‘Thing’ is NP. Nevertheless, NPs can also realize *Q* arguments, hence the ‘concealed question’ NP in (1b).

The ultimate constraint, however, stems from the interplay between the two sets of restrictions. Thus, although *Q* arguments *can* be realized as CPs or NPs, their actual realization depends on the syntactic subcategorization of the predicate. With *ask*, the two potential realizations of *Q* are possible, while the subcategorization property of *wonder* restricts it to CP. This accounts for the data in (1).

Grimshaw focuses on interrogative and exclamative complements, yet her claim regarding the independence of syntactic and semantic selection is general. A fully fleshed-out theory of argument structure in the spirit of Grimshaw must define a complete set of syntactic categories, a complete set of semantic categories and the CSR of each semantic category, as well as a set of its alternate realizations (e.g., NP as *Q*). These building blocks should be sufficient to determine the semantic selection and syntactic subcategorization of all predicates in the language, as well as the linking between the semantic selection and the syntactic subcategorization.

The relationship between the syntactic and semantic aspects of argument selection features in a study by Jackendoff (1985), which discusses the phenomenon of *multiple subcategorization*, whereby a predicate can appear in a number of different subcategorization frames. Jackendoff focuses on the verb *climb*, which can appear in three different configurations:

(2) (a) [__]: Bill climbed.
  (b) [__ NP]: Bill climbed the ladder.
  (c) [__ PP]: Bill climbed along the roof.

The sentences above are each prefixed with their respective subcategorization frame, represented in the notation originally proposed by Emonds (1991). The lines in this notation indicate the position of the verb, and the NP and PP indicate the syntactic category of the complements.

Bearing in mind that the three configurations all pertain to one lexeme, a more compact and telling notation could be used, with parentheses indicating the optionality of the complement:

(3) [____ (NP)(PP)]

However, as Jackendoff notes, this representation will not rule out the following ungrammatical sentence, where the two complements are realized:[3]

[3] Jackendoff concedes that a similar *Bill climbed the mountain up a narrow path* may be acceptable, and argues that the PP is non-subcategorized (based on extraction data), yet is semantically coordinated with the subcategorized NP. Nevertheless, mutual exclusivity of selected arguments does occur and it is discussed in Section 3.2.2.

(4) *Bill climbed the mountain up the rope.

A possible solution would be to use the set notation to indicate that only one complement is possible:

(5) [____ ({NP,PP})]

Grouping the NP and PP complement together, Jackendoff claims, misses a semantic distinction that differentiates between the two complement types. In both cases, the verb *climb* indicates motion. With the intransitive frame and the NP complement frame, the motion is understood as upwards (cf. (2a) and (2b)). With the PP complement, the direction of the motion is specified by the preposition heading the PP (2c). All of this information is part of the speaker's knowledge of the predicate.

The question, then, is how to capture the correspondence between syntactic arguments and semantic relations. Gruber (1965) was the first to refer to the semantic relations between predicates and their arguments as *thematic roles*. Subsequently, different sets of thematic roles (also known as *theta roles* or *θ-roles*) have been proposed in the literature.[4] The most common roles adopted in the literature are Agent, Patient, Theme, Location, Source, and Goal.

Nevertheless, in his discussion of cases such as the verb *climb*, Jackendoff (1985, 1987) contends that thematic roles cannot accurately capture the semantic relations between predicates and arguments. For example, it is plausible to assume that the semantics of *climb* involves a Goal. Nevertheless, it is the *top* of the ladder in (2b) and not the ladder itself, that is the Goal. Thus, Jackendoff claims, attributing this thematic role to the complement of the verb is not adequate. More generally, Jackendoff concludes, thematic roles are not primitives of semantic theory, and thus a more fine-grained semantic formalism is required.

This kind of phenomenon and argumentation led Jackendoff (1985, 1987) to propose an alternative theory, *Conceptual Semantics*. A basic assumption in this theory is that the semantic structure of a sentence is built up from *conceptual primitive* (Jackendoff 1987). One type of primitive is 'semantic parts of speech' such as Thing, Place, Path, Event, State, Manner, and Property. Correspondence rules map semantic parts of speech to syntactic ones. Each conceptual constituent has a syntactic phrase type which is its unmarked realization.[5] Thus, for example, the unmarked realization of Thing is NP, of Event is S, and of Property is AdjP. However, there are also marked realizations: Property can be realized by PPs (e.g., *out of luck*) or NPs (e.g., *bummer*).

Atomic semantic expressions can be expanded into more complex expressions by the application of semantic functions such as GO, BE, STAY, TO, FROM, and TOWARD. Thus, for example, a Path can be expressed by applying the TO

[4] See the discussion in Section 2.1.
[5] A similar idea is the Canonical Structural Realization (CSR) assumed by Grimshaw (1979).

function on a Thing such as *the house*. Following is an example of a conceptual structure and its corresponding syntactic structure.

(6) (a) $[_{Path}$ TO $([_{Thing}$ HOUSE$])]$
(b) $[_{PP}$ to $([_{NP}$ the house$])]$

An important principle of Conceptual Semantics is that there is no assumption of one-to-one correspondence between syntax and semantics. While this correspondence holds in the case of *to the house* in (6), this is not necessarily so. A case in point is the lexical entry that Jackendoff proposes for *climb*. The lexical entry expresses the fact that the verb denotes a motion event by specifying a semantic function, GO, and a Path specification. The Path specification is realized in two different ways, depending on the complement. The PP complement provides the total semantic content of Path in a compositional fashion, similarly to *to the house*. However, when the complement is an NP, its Path denotation is implicit and is expressed by the use of primitive semantic functions, as is illustrated below.

(7) $[_{Path}$ TO TOP OF $([_{Thing}$ LADDER$])]$

Syntactically there is only one argument, and it denotes a Thing. The fact that the Path is directed towards the *top* of the Thing is not expressed explicitly in the syntax. Rather, it is incorporated into the meaning of the verb *climb*.

The exact notation and specifics of the lexical entry are immaterial for our purpose.[6] Nevertheless, the point that we mean to stress is the observation that a formal syntax-only representation of multiple subcategorization is not sufficient to tell the full story. Rather, the interplay between syntax and semantics is a major factor in understanding argument structure. This observation guides the solution that we propose in Section 3.

Our solution distinguishes between syntactic selection and semantic selection. Consequently, constraints can be defined for each level individually, yet their interaction ultimately accounts for the argument structure of predicates. This, as we show, extends the expressive power of the mostly syntactically based Head-driven Phrase Structure Grammar (HPSG, Pollard & Sag 1994) approach to argument structure. As a result, generalizations that are currently overlooked can be made explicit in the grammar. Moreover, the analyses produced by the grammar are semantically informative and can serve as input to natural language processing tasks which require information related to the syntax *and* the semantics of linguistic expressions, such as machine translation (e.g. Bond et al. 2011), intelligent text understanding, etc.

We explore in Section 2 a number of different approaches for capturing the semantic relations that hold between predicates and their arguments. This serves as background for Section 3, where we present our proposal and illustrate its

[6] Incidentally, Butt (1995) uses Conceptual Semantics to formulate an analysis of Urdu complex predicates in Lexical Functional Grammar.

benefits by considering various phenomena that our proposal facilitates better, more general accounts for. In Section 4 we delve into more technical HPSG details, describing how our proposal is implemented in an LKB-based (Copestake 1999, 2002b) computational grammar of Modern Hebrew. We conclude with directions for future research.

2. Semantic selection

The previous section reviewed the theories of Grimshaw (1979) and Jackendoff (1985) regarding the relationship between syntax and semantics in the context of predicates and their argument structure. The two theories share a general view, namely that both semantic and syntactic categories are needed in order to account for the argument structure phenomena found in natural language. Furthermore, in both theories the relationships between the two domains are subject to correspondence rules. The main question, then, is what constitutes a possibly universal, finite, exhaustive, and well-defined set of semantic categories that can capture the semantic relationships between selecting predicates and their arguments. This is still an open question which has received many different answers over the last fifty years.

Dowty (1991) distinguishes between two types of understandings of thematic roles. The approach adopted by Jackendoff and Grimshaw views thematic relations as notions of conceptual structure, existing independently of syntactic or interface notions. A different understanding is referred to by Dowty as the 'argument-indexing' view of thematic roles and is tightly connected to syntax.

A strong formulation of the argument-indexing view is found in the θ Criterion (Chomsky 1981):

(8) *θ Criterion*
Each argument bears one and only one θ-role, and each θ-role is assigned to one and only one argument.

While there is no consensus regarding the first clause of the θ criterion, the principle expressed in the second clause figures in all the approaches we review.[7] Thus, regardless of the number or content of the semantic roles proposed in each approach, the principle of 'one argument per role' is maintained.

In what follows we review some of these approaches and discuss their applicability to the purpose of this study.

[7] The first clause of the θ criterion plays a crucial role in the analysis of control constructions such as *John wants to leave*. *John* in this sentence is both the 'wanter' and the 'leaver', in violation of the requirement that each argument bear only one θ role. In order to respect this constraint, the analysis of control in the Government and Binding framework (Chomsky 1981) posits a phonologically empty PRO as the subject of *leave*, and as the bearer of its semantic role. A different analysis is proposed in the HPSG framework (Sag & Pollard 1991), which does not assume the first clause of the principle.

2.1 *Fillmore's deep cases*

Fillmore's seminal paper 'The Case for Case' set the stage for all subsequent work on semantic roles. Inspired by Gruber (1965), Fillmore (1968) makes one of the first attempts to establish a list of semantic roles, or, as Fillmore referred to them, 'deep cases'. The list, as envisioned by Fillmore, is 'a set of universal, presumably innate concepts which identify certain types of judgments human beings are capable of making about the events that are going on around them, judgments about such matters as who did it, who it happened to, and what got changed' (Fillmore 1968: 24). Fillmore identifies an initial set of six deep cases:

Agentive the perceived instigator of the action, typically animate.

Instrumental the inanimate force or object causally involved in the action or state.

Dative the animate being affected by the state or action.

Factitive the object or being resulting from the action or state, or understood as a part of the meaning of the verb.

Locative the location or spatial orientation of the state or action.

Objective semantically most neutral, anything whose role is identified by the semantic interpretation of the verb itself.

However, he also notes that 'Additional cases will surely be needed. . .' (Fillmore 1968: 46–47), and subsequently proceeds to add Benefactive and Time, in his discussion of prepositions, and Comitative in the context of coordination.

Fillmore (1971) revises the list of semantic roles: the Dative role is replaced with the new Experiencer case, Locative case is split into three cases (Location, Source, and Goal), Factitive is subsumed under Goal, and Comitative is dropped. The result is a nine-case system (Agentive, Instrumental, Experiencer, Object, Location, Source, Goal, Time, Benefactive).

This early attempt at providing a comprehensive analysis of the semantic relations between predicates and their arguments has since taken different directions by different researchers. Broadly speaking, some approaches have extended the roleset and enriched the semantic representation of argument structure, while others, mostly computationally oriented, have attempted to reduce the set. In the following sections we describe a number of alternative approaches, and discuss whether they are suitable for representing the syntax–semantics relations in a wide-coverage computational grammar.

2.2 *Semantic roles in HPSG*

A number of different approaches to semantic roles are found in the HPSG framework. We describe below three approaches which vary in the level of generalizations they express and in the richness of their semantic representation.

2.2.1 *The original HPSG analysis*

The semantic relations assumed by Pollard & Sag (1994: 29) consist of the feature RELATION, whose value is atomic (e.g., *love*), and a number of predicate-specific role features (e.g., *lover*, *loved*). With relation-specific role labels, the total number of unique roles is proportional to the size of the lexicon. Moreover, an obvious shortcoming of this approach is that it does not provide any means of expressing generalizations regarding the arguments and the association between semantic roles and syntax.

Acknowledging this, the authors propose an alternative approach. They sketch a solution to the generalizability problem by suggesting (in their Section 8.5) to define a type hierarchy of relations, where shared role features are introduced for non-maximal relation types (i.e., more general types which dominate subtypes), and are ultimately shared by their subtypes. A sketch of this hierarchy, which is dominated by the general *quantifier-free parameterized states of affairs* (*qfpsoa*) type, is shown in Figure 1.

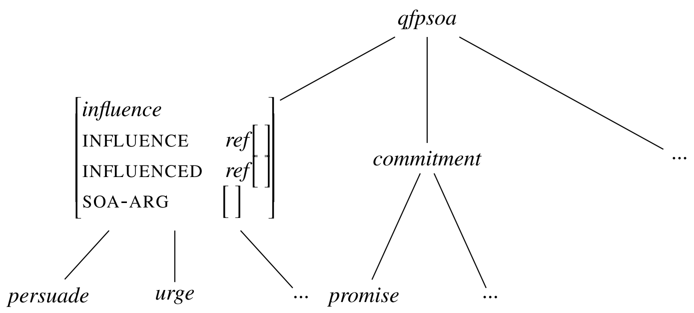

Figure 1
Semantic relations hierarchy (Pollard & Sag 1994).

With such a hierarchy, instead of individually defining for each verb its semantic relation, verbs with a similar semantic structure inherit from a general semantic relation type, for whom the shared features are defined. For example, the semantic relations denoted by verbs expressing influence (e.g., *persuade*, *urge*) will all be subtypes of a more general *influence* relation, and as such their semantic relation will include three semantic roles: Influence, Influenced and State-of-Affairs (SOA). A similar conceptualization of general semantic relations that are inherited by specific lexical items is found in the FrameNet project (see Section 2.4).

2.2.2 *Linking as constraints on word classes*

The idea of defining a hierarchy of semantic relations is further developed by Davis & Koenig (2000), who use general semantic categories such as Actor, Undergoer, and State-of-Affairs (SOA) to express more complex semantic relations. A sketch of this hierarchy is given in Figure 2 (Davis & Koenig's Figure 5). In this hierarchy, each supertype introduces one semantic role. More complex relations inherit feature specifications through multiple inheritance from multiple supertypes.

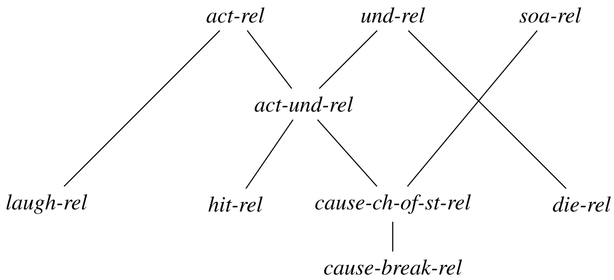

Figure 2
A hierarchy of semantic relations (Davis & Koenig 2000).

Consider as an example the *act-und-rel* relation, which includes two semantic roles, ACT(OR) and UND(ERGOER), which it inherits from the general types *act-rel* and *und-rel*, respectively. This relation in turn dominates instances of semantic relations, such as *hit-rel*. A more complex relation, *cause-change-of-state-rel*, inherits from the *act-und-rel* relation type and from *soa-rel*, which introduces the feature SOA, whose value is a relation which denotes the state caused by the action of the Actor.[8] Instances of this semantic class are the relations denoted by *cut* and *break*. Examples of the general *cause-change-of-state-rel* relation type and the specific *cause-break-rel* are illustrated in Figure 3.

$$\begin{bmatrix} \textit{cause-ch-of-st-rel} & \\ \text{ACTOR} & \textit{content} \\ \text{UNDERGOER} & \textit{nom-obj}\ \boxed{1} \\ \text{SOA} & \begin{bmatrix} \textit{rel} \\ \text{ARG}\ \boxed{1} \end{bmatrix} \end{bmatrix} \qquad \begin{bmatrix} \textit{cause-break-rel} & \\ \text{ACTOR} & \textit{content} \\ \text{UNDERGOER} & \textit{nom-obj}\ \boxed{1} \\ \text{SOA} & \begin{bmatrix} \textit{broken-rel} \\ \text{P-BEARER}\ \boxed{1} \end{bmatrix} \end{bmatrix}$$

Figure 3
General and specific semantic relations (Davis & Koenig 2000).

The main purpose of the semantic representation that Davis & Koenig (2000) propose is to account for linking patterns, that is, the systematic mapping between semantic arguments and syntactic functions. The semantic relation type hierarchy

[8] In a later paper, Koenig & Davis (2006) argue against this representation of semantic content, and propose a less recursive approach to semantic representation, where complex relations are represented as sets of relations.

that is illustrated in Figure 2 is mirrored by a parallel word-class hierarchy, where linking is defined. For example, the *act-und-rel* semantic relation type is associated with the *act-und-vb* verb type, where the Actor argument is linked to the subject and the Undergoer argument to the object. Since accounting for linking is the goal of the study, the set of semantic relations that Davis & Koenig assume are those that affect just that. Consequently, the set is limited to a few roles: Actor, Undergoer, SOA, Figure, Ground, Property-bearer. The authors demonstrate how linking patterns of various verb types and constructions that are considered a challenge for linking can be accounted for based on this set of roles.

A primary concern in this context is the lack of correspondence between syntactic arguments and semantic arguments. For Davis & Koenig (2000) this is an important feature of their proposal, as they argue that semantic representations should not be reduced to 'syntactic diacritics', to borrow their term. Consequently, they do not assume one-to-one correspondence between semantic roles and syntactic arguments. For instance, although cause-change-of-state verbs have two syntactic arguments, the semantic relation that they denote includes an additional role, SOA, which denotes the end-state and which is an implicit part of the content of the verb.[9] This is shown in Figure 3, where the end-state of the verb *break* is represented by the *broken-rel* relation, which does not have a syntactic counterpart. Conversely, the referent of the NP complement assumes two semantic roles: the Undergoer of the 'main' relation, and the Property-bearer of the embedded relation.

On the one hand, the system proposed by Davis & Koenig (2000) captures fine-grained semantic distinctions. For example, it characterizes the difference between two transitive verbs such as *hit* and *break*, whose subcategorization frames are identical. The semantic difference between the two is that the former has a simple actor–undergoer denotatum, while the latter is assumed to have a more complex caused-change-of-state semantics. On the other hand, although the inventory of semantic relations may be sufficient to account for non-trivial linking phenomena, it is quite limited. For example, their semantic representation does not distinguish between basic semantic roles such as Source and Goal (cf. Fillmore 1968). Consequently, it is not clear how scalable this schema is; the set of semantic roles may not be sufficient to account for large and diverse data, and, furthermore, the creation of a wide-coverage lexicon with such sophisticated representations is not a simple task.

[9] Note that a similar point is made by Jackendoff (1985) in his analysis of *climb*, whereby the semantic relation denoted by the NP complement includes semantic primitives (TO TOP OF) which do not have a syntactic counterpart. Similarly, Jackendoff (1987) argues that although the verb *to butter* is syntactically transitive (e.g., *Harry buttered the bread*), conceptually, in addition to the Actor (*Harry*) and the Goal (*the bread*), there is an implicit Theme argument, the butter, which is 'completely expressed by the verb' (p. 387).

2.2.3 *The* DELPH-IN *approach*

HPSG grammars implemented in the framework of the DELPH-IN initiative, such as the English Resource Grammar (Flickinger 2000), use Minimal Recursion Semantics (MRS; Copestake et al. 2005). MRS adopts a 'flat semantics' representation which allows for different levels of specification of scope (negation, quantifiers, adverbials, etc.). Underspecification is used to capture scopal ambiguities. The semantic roles in predicate–argument structure are represented in DELPH-IN grammars by what is referred to by Copestake (2009) as 'slacker semantics', in contrast to frameworks such as FrameNet (Section 2.4) which aim to provide elaborate, semantically rich representations. Slacker semantics, Copestake (2009) claims, is appropriate from a grammar engineering perspective.

Four general argument role features are used in feature structures of predicates: ARG1 through ARG4. Copestake (2009: fn. 5) notes that 'ARG4 occurs very rarely, at least in English (the verb *bet* being perhaps the clearest case)'. The approach is syntactically rather than semantically driven. Arguments are specified on the basis of the syntactic obliqueness hierarchy (Pollard & Sag 1994) and are assigned semantic roles consecutively. Thus, beginning with the least oblique argument (the subject in non-passivized predicates), arguments are assigned numbered roles (ARG1 through ARG4). There is no predicate that is specified for an ARG_n argument without selecting for ARG_{n-1}. Nevertheless, any argument may be syntactically optional, and thus not necessarily realized in the syntax. Moreover, the roles of predicates are relative to a predicate and remain constant across valence alternations. Consequently, active/passive and double object/dative pairs will have the same predicate–argument structure.

The main shortcoming of this system is that across predicates there is no semantic significance to the semantic labels with which arguments are associated. For example, being an ARG2 argument simply means that the argument is the second least oblique argument in the argument structure of the verb, regardless of whether it denotes a Theme, Goal, or Attribute (9).

(9) (a) Kim ate an apple.
(b) Kim flew to San Francisco.
(c) Kim seems happy.

Moreover, the class of syntactic phrases that realize a semantic role, say ARG2, most likely contains the entire inventory of syntactic phrases in the grammar. There is no principled restriction on the syntactic or semantic properties of the argument bearing this semantic role.

2.3 *PropBank*

A similarly limited set of semantic roles is used in the *PropBank* project, whose goal is to add a semantic layer to the syntactic trees of the Penn Treebank (Kingsbury & Palmer 2003). The semantic representation is therefore closely tied to the syntactic structure. The annotation scheme employed by PropBank attempts to maintain a small set of five roles, while providing consistent argument labels across different syntactic realizations of a verb sense. The scheme uses numbered roles due to a lack of consensus regarding semantic role labels, as well as the desire to limit the number of labels. Table 2 presents PropBank's set of role labels and corresponding semantic roles.[10]

Arg0	agent [also: experiencer]
Arg1	patient [also: theme]
Arg2	instrument, benefactive, attribute [also: end state]
Arg3	starting point, benefactive, attribute [also: instrument]
Arg4	ending point

Table 2
PropBank's roleset.

The annotation of Arg0 and Arg1 across predicates is consistent. Following Dowty (1991), the choice between Arg0 and Arg1 is made by comparing the number of proto-Agent and proto-Patient entailments that are valid for an argument. The greatest number of entailments determines whether the argument is labeled Arg0 (Proto-Agent) or Arg1 (Proto-Patient). The distinction between roles Arg2, Arg3, and Arg4 is verb-specific, yet is consistent across different syntactic realizations of the same verb sense. The lack of consistency across verbs is an inevitable consequence of the desire to define a small set of roles, while observing the principle of 'one argument per role'.

The annotation scheme provides a way of distinguishing among different senses of a given verb. For example, the verb *leave* has a number of senses, of which two are 'move away from' and 'give'. Each one of the senses is associated with its own roleset, where each role is explicated by the use of a sense-specific role, such as 'place left' or 'thing given'.

(10) (a) Roleset leave.01 'move away from':
Arg0: entity leaving
Arg1: place left
Arg2: attribute of Arg1

(b) Roleset leave.02 'give':
Arg0: giver
Arg1: thing given
Arg2: beneficiary

[10] The descriptions of the different roles are taken from the English PropBank Annotation Guidelines (Bonial et al. 2012), which are based on Palmer, Gildea & Kingsbury (2005). In square brackets are additional semantic roles, which appear in the PropBank tutorial given by Martha Palmer on June 9th 2013. Retrieved from http://naacl2013.naacl.org/Documents/semantic-role-labeling-part-1-naacl-2013-tutorial.pdf, January 2015.

In addition to the descriptions of the different rolesets, PropBank provides examples of the different syntactic argument frames in which the verb in a particular sense appears. Following are examples of two syntactic argument frames in which *leave* in its 'give' sense is found: the ditransitive frame (11a) and the double object frame (11b).

(11) (a) John left [$_{Arg1}$ cookies] [$_{Arg2}$ for Mary].
(b) John left [Mary $_{Arg2}$] [$_{Arg1}$ a collection of run-down clocks].

Consistency in annotation across predicate sense is maintained, as *(for) Mary* is labeled as Arg2, regardless of its syntactic realization and its position with respect to the other complement.

The PropBank schema is more semantically motivated than the 'slacker semantics' approach of DELPH-IN grammars, in that each of its five roles is associated with a number of common semantic roles, and a sense-specific characterization of these roles is defined specifically for each lexical entry. There is, however, no attempt to maintain consistent labels across different verbs and even verb senses. For example, Arg2 represents very different semantic roles in the two senses of the verb *leave*, illustrated in (10) above. Moreover, there is considerable overlap between the semantic roles associated with each label, as can be seen in Table 2. Thus, for example, Arg2 and Arg3 are associated with 'Benefactives' and 'Attributes'. This inconsistency is evident in the English framesets illustrated on the PropBank website for the predicates *give*,[11] *bake*,[12] and *buy*.[13]

(12) (a) [$_{Arg0}$ The executives] gave [$_{Arg2}$ the chefs] [$_{Arg1}$ a standing ovation].
(b) [$_{Arg0}$ John] baked [$_{Arg3}$ Mary] [$_{Arg1}$ a cake].
(c) [$_{Arg0}$ John] bought [$_{Arg4}$ his mother] [$_{Arg1}$ a dozen roses].

Although *the chefs*, *Mary*, and *his mother* are all the recipient/benefactive arguments of their respective verbs, they are labeled differently for each verb. Thus, their labels are only meaningful in the context of the particular roleset in which they occur, where their actual meaning is listed. As was previously mentioned, Arg0 and Arg1 are exceptions to this, as they are used fairly consistently across the predicates.

2.4 *FrameNet*

Semantically rich representations are proposed by FrameNet and VerbNet (Section 2.5), two lexical databases which contain detailed syntactic–semantic descriptions of lexical items, as well as annotated examples of how the words are used. The goal of the two projects is much greater than to provide a set of

[11] http://verbs.colorado.edu/propbank/framesets-english/give-v.html, retrieved January 2015.
[12] http://verbs.colorado.edu/propbank/framesets-english/bake-v.html, retrieved January 2015.
[13] http://verbs.colorado.edu/propbank/framesets-english/buy-v.html, retrieved January 2015.

semantic roles which capture the semantic relations between predicates and their arguments and which can be used to represent these relations in a computational grammar. Nevertheless, the identification and categorizations of these relations is an essential part of the two projects, and for this reason they are relevant to this study.

FrameNet is based on a theory of meaning called *Frame Semantics* (Fillmore 1982), which evolved from Fillmore's theory of deep cases (or Case Grammar), described in Section 2.1. The first step in this evolution was the realization that cases can be used to define situation types. Thus, for example, a 'caused change' situation is associated with an AIO (Agent–Instrument–Object) case frame (e.g., *I fixed it with a screwdriver*). Consequently, a large number of situation types were defined, which ultimately led to a new conceptualization: 'making frames primary, and defining roles in terms of the frames' (Fillmore 2012: 711), or, in other words, Frame Semantics.

The FrameNet project (Fillmore, Johnson & Petruck 2003) involves the construction of a database which includes a list of Frames. Fillmore (2012) reports that FrameNet has 1200 frames. Each frame is associated with *Frame Elements* (FEs), which are semantic labels for 'things worth talking about when a given frame is relevant' (p. 714). The list of FEs associated with a frame is divided into *core* FEs, which are required by the frame, and *peripheral* FEs, which are traditionally described as *adjuncts*. In addition, FrameNet includes a list of 13,000 lexical units – nouns, verbs, and adjectives – which are associated with frames. Each lexical unit appears with a set of annotated corpus-based sentences which illustrate the different ways in which the FEs can be realized.

As an illustration, consider the COMPLIANCE frame, its FEs, lexical units, and annotated sentences (Fillmore 2012).

(13) *Compliance*
This frame concerns [Acts] and [States of Affairs] for which [Protagonists] are responsible and which either follow or violate some set of rules or [Norms].

Each of the boxed expressions in the definition is an FE. Among the lexical units associated with COMPLIANCE are the positive *abide*, *observe*, and *obey* and the negative *breach*, *flout*, and *violate*. Following is an example of annotated sentences with the lexical unit *violate*:

(14) (a) A lot of people suspect that [$_{ACT}$ blocking savers' access to their money] violates [$_{NORM}$ the new constitution's explicit ban on compulsory loans to the government].
(b) He is still guilty because [$_{PROTAGONIST}$ he] has violated [$_{NORM}$ God's law].
(c) This is not because [$_{STATE\ OF\ AFFAIRS}$ the law] does violate [$_{NORM}$ human rights].

Similar FE labels may be found in different frames. Thus, for example, there could be other frames with a Protagonist. Nevertheless, the theory assumes that FEs are local, or, in other words, only relative to the frame in which they appear. For this reason, it is meaningless to evaluate the number of roles used by FrameNet. Moreover, it is not possible to isolate FrameNet's semantic roles from the entire framework. Thus, this approach is not suitable for our goal.

2.5 *VerbNet*

The structure of the *VerbNet* lexicon (Schuler 2005) is based on Levin's (1993) verb classification. According to Levin, verbs can be classified according to the types of syntactic alternations in which they can appear. Consequently, entries in VerbNet are associated with a particular verb class, and pertain to a number of verb members that share syntactic and semantic properties.

One well-known verb class is the *Spray/Load* class, which includes verbs that participate in the locative alternation.

(15) (a) Jessica sprayed paint on the wall.
(b) Jessica sprayed the wall with paint.

Some other verbs that belong to this class are *splash*, *scatter*, *drizzle*, and *sprinkle*.

VerbNet augments Levin's verb classes by (1) listing the thematic roles (e.g., Agent, Theme, Destination) associated with the verb class, (2) assigning a thematic role to each syntactic argument in each argument frame, and (3) describing the meaning of the frame by using semantic primitives (e.g., Motion, Location, Cause). In addition, when relevant, thematic roles appear with selectional restrictions which address the existence or absence of properties such as CONCRETE, SUBSTANCE, ANIMATE, or ORGANIZATION.

Two frames are associated with the *Spray* class, each pertaining to one of the alternates illustrated in (15) above.

SYNTAX	NP.AGENT V NP.THEME PP.DESTINATION
SEMANTICS	MOTION(DURING(E), THEME) NOT(PREP(START(E), THEME, DESTINATION)) PREP(END(E), THEME, DESTINATION) CAUSE(AGENT, E)

SYNTAX	NP.AGENT V NP.DESTINATION PP.THEME
SEMANTICS	MOTION(DURING(E), THEME) NOT(LOCATION(START(E), THEME, DESTINATION)) LOCATION(END(E), THEME, DESTINATION) CAUSE(AGENT, E)

While both alternations share the same set of thematic roles, the syntactic realization of these roles is distinct (hence the alternation). Moreover, the semantics of the two alternations differ: whereas (15b) implies that the location/container is completely filled or covered, (15a) does not. This difference is expressed in the semantic representation in the respective frames.

Unlike FrameNet, VerbNet assumes a general list of semantic roles, which apply to all verb classes and verbs. Altogether, VerbNet's roleset includes 36 roles.

(16) Actor, Agent, Asset, Attribute, Beneficiary, Cause, Co-Agent, Co-Patient, Co-Theme, Destination, Duration, Experiencer, Extent, Final_Time, Frequency, Goal, Initial_Location, Initial_Time, Instrument, Location, Material, Participant, Patient, Pivot, Place, Product, Recipient, Result, Source, Stimulus, Time, Theme, Trajectory, Topic, Undergoer, Value.

The roles are defined in an inheritance hierarchy which is dominated by four general roles: Actor, Undergoer, Time, and Place. The Actor role dominates two more specific roles: Agent and Cause, distinguished by the negative and positive values of the feature INTENTIONAL. The Cause role further dominates the Stimulus role. More specific roles are often used for specific types of verb classes. Thus, for example, the Topic role, a sub-role of Theme, which in turn is dominated by Undergoer, is associated only with information transfer or communication verbs. Roles that are in a parent–child relationship cannot co-occur.

VerbNet's sizable roleset provides a way to represent the semantic relations between predicates and their arguments in a relatively fine-grained manner. Moreover, this roleset can be adopted as an independent component, without necessarily adopting Levin's classification of verbs. However, with fine-grained distinctions between semantic relations it is at times difficult to decide between a number of alternative relations (e.g., the precise difference between Undergoer and Patient). Indeed, Jaworski & Przepiórkowski (2014a) report low inter-annotator agreement regarding the use of VerbNet's roles in Polish. Consequently, with large-scale computational grammars, using a smaller, more broadly defined set of roles may be more feasible.

The hierarchical structure in which the roles are defined in VerbNet addresses the need for more general categories; a number of lower-level roles can be collapsed into one role, the one that dominates them. This, however, has proven to be quite tricky, since the hierarchy includes cases of multi-inheritance where a given role is dominated by more than one role (a case in point is Result, which is dominated by both Goal and Patient[+affected], where the former is dominated by the general role Place, and the latter by Undergoer). Consequently, there is no straightforward way to extract a smaller, more general set of roles from the VerbNet roleset.

2.6 *Syntactic approximation of semantic roles*

Jaworski & Przepiórkowski (2014a) experimented with adopting VerbNet semantic roles for Polish. Seven annotators used VerbNet's set of roles to annotate sentences containing 37 randomly selected verbs. Altogether there were 393 occurrences of the verbs, with a total of 843 arguments. An analysis of the annotations revealed low inter-annotator agreement (κ score of 0.617).[14] The two main reasons the authors give for this low score are (1) numerous cases where more than one argument role seemed to fit and (2) cases where no suitable role was found. They conclude that VerbNet's schema may not be suitable for their grammar engineering task, which is to add a semantic component to a Lexical Functional Grammar (LFG, Kaplan & Bresnan 1982) of Polish.

As an alternative, Jaworski & Przepiórkowski (2014b) propose a method for assigning semantic roles to arguments in languages that have rich morphosyntax. They exploit the rich morphological case system of Polish (with seven cases) in order to define a small set of 11 semantic roles, or labels (see Table 3). Each category is associated with a number of prepositions and/or morphological cases. The authors concede that these categories are only approximations of semantic roles, and consequently refer to them as 'semantic roles' (in scare quotes) and name them R0, R1, etc.

R0	Actor of an action (Agent, Effector)
R1	Undergoer of an action (Patient, Theme, Product)
R2	Dative argument (Beneficiary, Recipient)
R3	Instrumental argument (Instrument)
R4	Adlative argument in both physical and abstract (functional, purposive) meaning (Destination, Recipient, Theme)
R5	Ablative argument in both physical and abstract (causal) meaning (Source)
R6	Locative argument in both physical and abstract meaning
R7	Perlative argument
R8	Topic of communication
R9	Temporal argument (point in time)
R10	Manner argument

Table 3
Jaworski & Przepiórkowski's 'semantic roles'.

This semantic roleset proposed by Jaworski & Przepiórkowski (2014b) is richer than the rolesets used by PropBank and in computational HPSG, and it allows for consistent labeling across predicates. While semantically coherent, it is strictly 'argument indexing', in that it does not introduce implicit semantic features. Moreover, each preposition or morphological case is matched with only one role, thus making the labeling process quite deterministic.[15] However, due to the

[14] Similar results were also found with a schema by Sowa (2000), which we do not discuss here.

[15] Jaworski & Przepiórkowski (2014b) report that in Walenty, a valence dictionary for Polish (Przepiórkowski et al. 2014), only 1.42% morphosyntactic schemata contained two or more arguments that would be mapped to the same role. In almost half of the cases this was the result of a verb selecting a number of prepositional arguments of the same type.

nature of the task, this schema is closely tied to the Polish language with its rich morphosyntactic system and what Jaworski & Przepiórkowski (2014b) view as a semantically coherent use of prepositions. As such, it may not be appropriate for representing semantic relations across languages. For example, R10 'Manner Argument' is associated with a single Polish preposition, *według* ('*according to*'), which does not seem to correspond to a core semantic role in Hebrew. Nevertheless, as was mentioned at the beginning of this section, the question of whether there is a possibly universal, finite, exhaustive, and well-defined set of semantic categories remains an open question.

2.7 *Conclusion*

This section presented a number of different approaches to analyzing and representing the semantic relations between predicates and their arguments. One significant difference between the aforementioned schemas for the semantic categorization of arguments lies in the tension between two desiderata:

- Coherent, semantically driven argument labels.
- A small, manageable set of argument labels.

While the motivation behind each of these desiderata is clear, one cannot ignore their conflicting nature. Consequently, we can identify a cline, ranging from the semantically rich approach adopted by FrameNet (Fillmore et al. 2003), whereby semantic labels are frame-specific and frames are organized in a large and intricate hierarchy, to the minimalist approach employed in DELPH-IN grammars (e.g., The English Resource Grammar; Flickinger 2000), where a small set of four consecutively numbered labels (Arg1, Arg2, . . .) is assigned to arguments. The remaining approaches occupy different positions along this cline.

Moreover, a crucial factor which distinguishes between the various approaches is the extent to which the semantic representation is dependent on the syntactic structure. As mentioned, Dowty (1991) distinguishes between conceptual approaches and 'argument indexing' approaches. However, even within the latter approaches, there are varying degrees of overlap between syntax and semantics in the domain of argument structure. The schemas adopted by PropBank and Jaworski & Przepiórkowski (2014b) are designed to add a semantic component to syntactic representations (Penn Treebank and LFG f-structures, respectively). As such, these schemas are more tightly connected to syntax, yet they are not involved in the parsing process. Conversely, Davis & Koenig (2000) argue that a true semantic representation cannot be reduced to 'syntactic diacritics', and consequently in their system there is no one-to-one correspondence between semantic roles and syntactic arguments.

The representation of argument structure that we propose is designed to capture the syntactic and semantic aspects of argument structure while striking the right balance between the two levels. As such, our approach is closest in spirit to that proposed by Jaworski & Przepiórkowski (2014b). It is syntactically driven, in that the semantic representation is directly related to the syntactic structure. Nevertheless, the inventory of semantic roles is sufficiently extensive to be meaningful, and is arguably universal. Moreover, as we show in Section 3.2, the approach that we propose extends the expressive power of the grammar by providing a way to state generalizations that are not captured by 'standard' HPSG (e.g., Sag, Wasow & Bender 2003).

3. Representing the syntax–semantics interface in argument structure

The relationship between predicates and their arguments involves both syntax and semantics. Following Grimshaw (1979), we view syntactic selection and semantic selection as two separate yet correlated phenomena. Representing this relationship involves accounting for each of the levels separately, as well as for the correspondences between the two levels. The rest of this paper is dedicated to doing just that.

Generally speaking, there is broad consensus regarding the types of syntactic categories that are involved in syntactic selection. The semantic side, however, as is evident from the previous discussion, has received many different analyses, with no one accepted approach.

The representation that we propose is guided by a number of principles:

- The semantic roles need to be semantically contentful and clearly defined.
- The semantic roles should be language-independent.
- The semantic roles should be consistent across all predicates, that is, a semantic label should have the same denotation regardless of the predicate.
- There should be one argument per semantic role.

As shown, such principles cannot be maintained with a very limited set of roles (e.g., four in DELPH-IN grammars, or five in PropBank). However, bearing in mind the cautionary words of Copestake (2009) (i.e., 'slacker semantics'), it is essential to maintain a realistic implementationally viable number of roles which will be compatible with the requirements of implementing a broad-coverage grammar. Thus, with respect to the richness of the semantic representation we attempt to find an optimal compromise between elaborate semantic representations and realistic grammar engineering.

Moreover, aside from the size of the roleset, an additional design decision stems from the nature of the task. Our main efforts in the implementation of the grammar are focused on accounting for the syntax of the language, not on implementing a semantic theory. Consequently, we adopt a 'pure' argument indexing approach, whereby semantic roles are necessarily linked to syntactic arguments. While this approach does not capture the fine-grained semantic distinctions that some of the semantics-centered approaches we reviewed express, the analyses produced by the grammar are semantically informative and potentially useful for tasks that require a deeper understanding of natural language.

Finally, the correspondence between the syntactic and semantic levels is captured in our system by the association of each semantic role with a class of syntactic categories that are potentially interchangeable with respect to the realization of that role. We will refer to them here as *realization classes*. This idea is close in spirit to Grimshaw's and Jackendoff's association of semantic primitives with syntactic categories. Yet, contrary to them, we do not assume unmarked or canonical structural realizations.

3.1 *Semantic roles and syntactic realization classes*

The semantic roleset is derived from the study of existing rolesets, described in Section 2, and corpus-based data. The corpus data include at least 100 randomly selected examples of sentences containing each of the 50 most frequent verb lemmas in the 60-million token WaCky corpus of Modern Hebrew (Baroni et al. 2009). First, the arguments of each verb in all its instances were identified.[16] Then, we attempted to label each argument with a semantic role, first considering the applicability of the roles that were proposed in the literature we reviewed. Consequently, there is considerable overlap between our roleset and existing rolesets. Most notably, the list proposed by Jaworski & Przepiórkowski (2014b) bears the most resemblance to our proposed list, since it conforms with most of our desiderata.

Following are the semantic categories we propose, along with their corresponding characterizations. We illustrate each category with an English example to simplify (and shorten) the presentation.

(*Arg*1**)** The prototypical member of this category is an *Actor*. Also included are Perceiver and Causer.

(17) (a) [$_{Arg1}$ John] walked home.
(b) [$_{Arg1}$ John] saw Mary.
(c) The book was written [$_{Arg1}$ by John].

(*Arg*2**)** This category corresponds to what is often referred to as *Theme*. Members of this category are negatively characterized as not belonging to the other, more semantically coherent, categories.

(18) (a) John wanted [$_{Arg2}$ a present].
(b) John wanted [$_{Arg2}$ to dance].

[16] Going into the issue of the argument–adjunct distinction is beyond the scope of this paper, but it has been discussed in the HPSG literature (Przepiórkowski 1999, Bouma, Malouf & Sag 2001).

(*Arg*3) Members of this category are prototypically animate beings which are positively or negatively affected by the action or state. It includes arguments that can be labeled *Affectee*, *Benefactive*, *Malfactive*, or *Recipient*.

(19) (a) John sent [$_{Arg3}$ Mary] a letter.
(b) John stole the idea [$_{Arg3}$ from Mary].

(*Arg*4) This category includes semantic arguments that are *Attributes*, or, in other words, predicative of another argument in the clause.

(20) (a) John found Mary [$_{Arg4}$ doing homework].
(b) John was [$_{Arg4}$ a teacher].

(*Arg*5) This category includes semantic arguments that denote *Source* in both its physical and metaphorical meanings.

(21) (a) John got a present [$_{Arg5}$ from Mary].
(b) John took the parcel [$_{Arg5}$ from his workplace].

(*Arg*6) This category includes semantic arguments that denote *Goal* in both its physical and metaphorical meanings.

(22) (a) The display changed from black and white [$_{Arg6}$ to color].
(b) John sent a letter [$_{Arg6}$ to Paris].

(*Arg*7) This category includes semantic arguments that denote *Location* in both its physical and metaphorical meanings.

(23) (a) John put the box [$_{Arg7}$ on the table].
(b) The decision was based [$_{Arg7}$ on new evidence].

(*Arg*8) This category includes arguments that denote the *Topic* (*of Communication*).

(24) The couple talked [$_{Arg8}$ about their trip].

(*Arg*9) Members of this category denote *Instrument* arguments, usually an inanimate entity causally involved in the action.

(25) John opened the door [$_{Arg9}$ with a key].

(*Arg*10) Members of this category denote *Comitative* arguments, usually an animate being who accompanies another participant of the event.

(26) John spoke [$_{Arg10}$ with Mary].

The coherence of this roleset was put to the test by having two annotators (one of whom was the original annotator) annotate a new set of 200 sentences, which included 10 sentences each for 10 verbs from the original frequent verb set, and 10 each for additional (new) verbs. The annotation of semantic roles was compared, and there was 86% (172/200) agreement between the two annotators, where 'agreement' was defined as annotating the exact same semantic roles. There were a few discrepancies, but all were easily resolved by a short discussion between the annotators. These include distinctions between complements and adjuncts, between unaccusative and unergative verbs, between Recipient (Arg3) and Goal (Arg6), etc.

The association of the arguments of the 50 most frequent verbs with the semantic roles described above revealed syntax–semantics correspondences, or, as we refer to them here, *realization classes*. The list of the semantic roles along with their realization classes is presented in Table 4, and the number of instances of each role is presented in Table 5.

Label	Semantics	Syntactic Realization
Arg1	Actor, Perceiver, Causer	NP, PP
Arg2	Theme	NP, VP_{inf}, CP, PP
Arg3	Affectee, Benefactive, Malfactive, Recipient	NP, PP
Arg4	Attribute	AdjP, AdvP, PP, NP, $VP_{beinoni}$[17]
Arg5	Source	PP
Arg6	Goal	PP
Arg7	Location	PP, AdvP
Arg8	Topic of Communication	PP
Arg9	Instrument	PP
Arg10	Comitative	PP

Table 4
Semantic roles and realization classes.

Arg1	Arg2	Arg3	Arg4	Arg5	Arg6	Arg7	Arg8	Arg9	Arg10
179	160	19	5	6	4	5	2	0	0

Table 5
The number of instances of each semantic role in 200 sentences.

[17] The 'beinoni' is a mixed category between verbs and nouns.

Each verb lexeme in the grammar is associated with two types of argument structure information: semantic frames and syntactic realization choices. Semantic frames are similar to Fillmore's *case frames*, which list the semantic arguments of the verb (e.g., AIO: Agent–Instrument–Object). In our schema they are expressed by referring to the numbered labels of semantic roles (e.g., *arg129* for Agent–Object–Instrument).[18] Corresponding to the semantic frame is a syntactic component which states for each semantic argument which syntactic phrase types are used to realize it. The choice of syntactic phrase types is restricted by the realization class associated with each semantic role.[19]

Whereas the semantic classes are expected to be more or less universal, some language-specific differences are predicted to be found in the syntactic realizations. Corpus investigations in the context of the development of AraGram, an HPSG grammar of Modern Standard Arabic, confirmed these expectations (Arad Greshler et al. 2015). The Arabic verbs that correspond to the 50 most frequent Hebrew verb lemmas were found to share the same semantic frames. Slight differences were found in the syntactic realization classes of Arg2 and Arg6.

As was shown by Grimshaw (1979), the two levels are crucial since one cannot be deduced from the other. NP complements may play different roles in their relations with their selecting predicate (e.g., Theme, Benefactive, Attribute). Conversely, semantic roles can be realized by phrases of different syntactic categories. Moreover, syntactic selection is largely lexeme-specific, yet constrained by the realization class of the respective semantic role (cf. Grimshaw's *ask* and *wonder*).

In order to illustrate the schema without going into HPSG-specific details, we use in this section a semi-formal notation to describe lexical entries. The description of each entry includes (1) the semantic arguments that the verb selects, (2) the realization class of each argument, (3) the realization frames (R-FRAMES) that the verb licenses. The latter property is indicated by a string such as *arg1/12/129*, where a slash appears between frames to indicate disjunction, and in each frame the numbers indicate which arguments are realized.[20]

For example, the lexical entry in (27) describes a predicate that selects three semantic arguments: Arg1, Arg2, and Arg9. Arg1 and Arg9 are realized by NP and PP, respectively. Arg2 can be realized by either NP, CP, or PP. Furthermore,

[18] The use of numbers instead of abbreviated role names makes the semantic frame names more compact.

[19] We are aware that the syntactic categories listed in Table 4 are too broad for predicates with more specific requirements, e.g., *wonder* in (1c) which selects only interrogative CPs. We intend to extend the infrastructure to account for such cases in the near future.

[20] The presentation here relates to relatively basic phenomena, but see Section 4.1 for a more detailed description of how the system handles cases such as raising, where a syntactic argument of a predicate is not its semantic argument.

the predicate can be realized in three different frames: all arguments are realized (*arg129*), Arg2 is omitted (*arg19*), or both Arg2 and Arg9 are omitted (*arg1*).[21]

(27)

R-FRAMES	*arg1/19/129*
ARG1	{NP}
ARG2	{NP, CP, PP}
ARG9	{PP}

Although this information is stated explicitly in our semi-formal lexical entries, in the grammar itself lexemes are instances of lexical types which are cross-classified according to the different argument slots and realization specifications. This type of architecture is similar in spirit to the one proposed by Davis & Koenig (2000), where more specific types multi-inherit from a number of more general types. Thus, similarly to Davis & Koenig's *act-und-rel*, which inherits from *act-rel* and *und-rel*, lexemes such as the one illustrated in (27) are instances of a lexical type that inherits from *arg1_n*, *arg2_ncp* and *arg9_p*,[22] each contributing argument-specific information. In a type inheritance hierarchy, generalizations can be stated once for types at the appropriate level, and this information is further inherited by all subtypes of that type. With multiple-inheritance, constraints regarding syntax and semantics combine together to account for the argument structure of lexical types.

3.2 *Benefits for linguistic analysis*

The proposed approach involves significant changes to the way in which argument structure is viewed. The precise HPSG-specific aspects of this system are discussed in Section 4, which describes the way in which it is implemented. In this section, however, we adopt a framework-neutral perspective.

The system that we propose here distinguishes between semantic selection and syntactic selection, and provides a way of stating constraints regarding each level separately. More specifically, with this system it is possible to define constraints that target Theme arguments, regardless of their syntactic category, or constraints that distinguish between PPs that realize Recipients and PPs that realize Goals. This extends the expressive power of the HPSG approach and provides a way to account for phenomena that are better characterized in semantic terms. The following sections illustrate some of these phenomena.

[21] We adopt a fairly liberal approach with regards to the complement–adjunct distinction, allowing adjunct-like dependents to be optional arguments. Nevertheless, phrases that are not selected are not associated with a particular role in the predication. They contribute to the semantic representation of the clause, but they do not add extra roles to the predication.

[22] The characters 'n', 'c', and 'p' stand for NP, CP, and PP, respectively.

3.2.1 *Multiple subcategorization*

Section 1 provided a short discussion of the phenomena that prompted Grimshaw (1979) and Jackendoff (1985) to consider the relationship between syntax and semantics in the domain of argument structure. In the two aforementioned studies the respective authors considered cases of multiple subcategorization and their implications for theories of argument structure.

The fact that predicates can appear in a number of different subcategorization frames is not difficult to capture. It is always possible to posit multiple lexical entries for a lexeme, each corresponding to one subcategorization frame. However, such a solution, we claim, overlooks important generalizations regarding the interchangeability of syntactic realizations of semantic arguments.

Consider as an example the three different subcategorization frames that are licensed by the verb *raca* ('*want*'):

(28) (a) dan raca kelev
Dan wanted dog
'Dan wanted a dog.'

(b) dan raca liqnot kelev
Dan wanted to.buy dog
'Dan wanted to buy a dog.'

(c) dan raca še-horav yiqnu lo kelev
dan wanted that-his.parents will.buy to.him dog
'Dan wanted his parents to buy him a dog.'

The complements of the verb denote whatever is wanted by the referent of the subject, regardless of whether the complement is a Thing, realized as an NP, or an Event, realized as a VP or CP. Thus, we posit that the three phrase types are members of one realization class which corresponds to the semantic role associated with the complement.

The same phrase types (i.e., NP, VP, CP) are also used as complements of the verb *lamad* ('*learn*'), as is shown below.

(29) (a) dan lamad nagarut
Dan learned carpentry
'Dan learned carpentry.'

(b) dan lamad livnot aronot
Dan learned to.build cabinets
'Dan learned to build cabinets.'

(c) dan lamad še-qaše livnot aronot
Dan learned that-hard to.build cabinets
'Dan learned that it is hard to build cabinets.'

The fact that these phrase types are found to be interchangeable as complements of various verbs suggests that they form a class. In our schema, this class is associated with the Theme argument role, annotated as Arg2. Under this type of analysis, the similar multiple subcategorization patterns observed for the two distinct verbs are not viewed as coincidental. Rather, they are captured by the schema, which assumes the correspondence between semantic roles and realization classes.[23] Consequently, the lexical entries of the two lexemes share the same argument structure specification.

(30) *raca* (‘*want*’), *lamad* (‘*learn*’):

$$\begin{bmatrix} \text{R-FRAMES} & \textit{arg12} \\ \text{ARG1} & \{\text{NP}\} \\ \text{ARG2} & \{\text{NP, VP, CP}\} \end{bmatrix}$$

3.2.2 *Argument optionality and co-occurrence restrictions*

It is well known that not all semantic arguments need to be realized syntactically. Argument optionality is usually indicated in the subcategorization notion shown in Table 1 above, and in (‘pen-and-paper’) phrase structure grammar by surrounding the phrase type with parentheses. This would suggest that argument optionality is a syntactic constraint. However, we argue that the actual level where constraints regarding the optionality of arguments should be stated is the semantic level. For example, the fact that the NP/CP/VP complement of the verb *raca* (‘*want*’) in sentences (28) above is obligatory need not be stated about each subcategorization frame separately. Rather, it is a general constraint about the Theme argument of this verb, *regardless of how it is realized*. Stating this separately for each subcategorization frame misses an important generalization.

Let us consider the following two example sentences of the verb *amar* (‘*tell*’), where optionality is indicated by parentheses.[24]

[23] This is not to say, of course, that all verbs that subcategorize for one member of a class subcategorize for the rest of the members. The analysis does claim, however, that Theme arguments will not be realized by an AdjP or an AdvP.

[24] While, as one reviewer pointed out, the PP ‘about this subject’ in (31a) can be construed as an argument of ‘his opinion’, it can appear in different positions, thus suggesting that it is an argument of the verb:

(i) (a)

[ʿal	ha-nose	ha-ze $_{Arg8}$]	[hu $_{Arg1}$]	amar	[le-kulam $_{Arg3}$]	[et	daʿto $_{Arg2}$]
about	the-subject	the-this	he	told	to-everyone	ACC	opinion.his

‘About this subject, he told everyone his opinion.’

(b)

[hu $_{Arg1}$]	amar	[et	daʿto $_{Arg2}$]	[le-kulam $_{Arg3}$]	[ʿal	ha-nose	ha-ze $_{Arg8}$]
he	told	ACC	opinion.his	to-everyone	about	the-subject	the-this

‘He told everyone his opinion about this subject.’

(31) (a) [hu $_{Arg1}$] amar ([le-kulam $_{Arg3}$]) [et daʿto $_{Arg2}$] ([ʿal
he told (to-everyone) ACC opinion.his (about
ha-nose ha-ze $_{Arg8}$])
the-subject the-this)
'He told (everyone) his opinion (about this subject).'

(b) [hu $_{Arg1}$] amar ([le-dan $_{Arg3}$]) [še-ani codeq$_{Arg2}$]
he told (to-Dan) that-I right
'He told (Dan) that I was right.'

We claim that the fact that the NP in (31a) and CP in (31b) are both not optional is not coincidental; they both realize the same semantic role, Theme, and this semantic argument is obligatory in the case of *amar* ('*tell*'). Conversely, Arg3, the Recipient, and Arg8, the Topic of Communication, are optional. This generalization can be simply stated in our system, as is evident from the argument structure specification of the lexical entry of *amar* ('*tell*') in (32); Arg2 is associated with the set containing both NP and CP, and all the disjoined frames in R-FRAMES contain '2'.

(32) *amar* ('*tell*'):

R-FRAMES	*arg12/123/128/1238*
ARG1	{NP}
ARG2	{NP, CP}
ARG3	{PP}
ARG8	{PP}

An additional phenomenon that our system provides the means to capture involves cases where the realization of one argument blocks the realization of another, although they both play different semantic roles. Consider the following example:[25]

(33) (a) hu x̱ašaš [mi-hefsed $_{Arg2}$] / [le-hefsed $_{Arg2}$] / [lehafsid $_{Arg2}$] /
he feared [from-loss] / [to-loss] / [to.lose] /
[še-hu yafsid $_{Arg2}$]
[that-he will.lose]
'He feared a loss / a loss / to lose / that he would lose.'

(b) hu x̱ašaš [le-ʿatid-o $_{Arg3}$]
he feared to-future-his
'He feared for his future.'

[25] We thank Edit Doron (p.c.) for this example.

(c) *hu x̲ašaš [le-ʿatid-o $_{Arg3}$] [mi-hefsed $_{Arg2}$]
he feared to-future-his from-loss

The complements of *x̲ašaš* (‘*fear*’) in (33a) all denote negative outcomes of which the referent of the subject is afraid, and each is realized by a different syntactic phrase type (PP_{mi}, PP_{le}, VP_{inf}, and CP, respectively). The complement of the verb in (33b), on the other hand, denotes the entity that can be badly affected by the negative outcome. Although the two arguments have clearly distinct semantic roles, and there does not seem to be a sense difference between the two *x̲ašaš* (‘*fear*’) in (33), these arguments cannot co-occur in the same clause, as is shown in (33c).[26]

This co-occurrence restriction can be stated once, as a semantic constraint on the co-occurrence of Arg2 and Arg3. More specifically, the verb *x̲ašaš* (‘*fear*’) is defined as semantically selecting three arguments (Arg1, Arg2, Arg3), yet its semantic realization frame specification is *arg12/13*, thus ruling out the co-occurrence of Arg2 and Arg3.

(34) *x̲ašaš* (‘*fear*’):

$$\begin{bmatrix} \text{R-FRAMES} & \textit{arg12/13} \\ \text{ARG1} & \{\text{NP}\} \\ \text{ARG2} & \{\text{PP, VP, CP}\} \\ \text{ARG3} & \{\text{PP}\} \end{bmatrix}$$

Here, too, the restriction is stated in terms of semantic selection, and is independent of the multiple syntactic realization options shown in (33a) for Arg2.

3.2.3 *Multiple subcategorization versus polysemy*

When syntactic and semantic constraints are conflated, multiple subcategorization and polysemy can be indistinguishable: both phenomena involve multiple lexical entries for a given lexeme. In our system we consider multiple subcategorization as a phenomenon where an identical semantic frame can be syntactically realized in different ways, either through argument optionality, or in terms of the use of different phrase types within the same realization class. As shown in the previous sections, this phenomenon can be defined once for one lexical entry. Polysemy, on

[26] Pesetsky (1996) discusses a similar co-occurrence constraint, which he refers to as the Target/Subject Matter (T/SM) restriction, according to which psych verbs cannot have both a Causer argument and a Target:

(i) (a) Bill was angry [$_{target}$ at the government].
(b) [$_{causer}$ The article in the Times] angered Bill.
(c) *[$_{causer}$ The article in the Times] angered Bill [$_{target}$ at the government].

the other hand, involves different senses, which may be associated with different semantic frames. In this case distinct lexical entries are defined, one per sense.

We will illustrate this point with the verb *amar* ('*tell*'). The sense denoted by the verb in (31) belongs to FrameNet's Telling frame, whereby 'A Speaker addresses an Addressee with a Message, which may be indirectly referred to as a Topic.' There is, however, a related sense of the verb, which is associated with a Request frame.[27] This is illustrated by the following sentence.

(35) [hu $_{Arg1}$] amar [le-kulam $_{Arg3}$] [lacet me-ha-xeder $_{Arg2}$]
he told to-everyone to.exit from-the-room
'He told everyone to get out of the room.'

The semantic frame of this sense contains three obligatory arguments: a Speaker, an Addressee, and a Message (in FrameNet terms). In our system, the Addressee is categorized as the Arg3 argument and the Message as Arg2. These three semantic arguments are shared between the two senses. However, unlike the Telling sense, the Request sense is incompatible with a Topic (Arg8) argument. Consequently, this sense is associated with a single semantic frame, *arg123*, and a syntactic constraint which indicates that Arg2 should be realized as VP_{inf} and Arg3 as PP. An additional characterization of this lexeme is the control pattern, whereby the unexpressed Agent (Arg1) of the VP complement is construed as (or controlled by) the referent of the object of *amar* ('*tell*'), the Arg3 argument.[28]

Consequently, only two lexical entries are needed in order to account for the multiple subcategorization as well as the polysemy of *amar* ('*tell*'); one entry per sense (36).

(36) *amar* ('*tell*'):

R-FRAMES	*arg12/123/128/1238*
ARG1	{NP}
ARG2	{NP, CP}
ARG3	{PP}
ARG8	{PP}

R-FRAMES	*arg123*
ARG1	{NP}
ARG2	{VP}
ARG3	{PP}

3.2.4 *NPs and their semantic role*

One aspect of the relationship between syntax and semantics in the domain of argument structure is expressed in the correlation between the morphosyntactic properties of NP complements, their syntactic function, and their semantic role. A case in point is their case marking patterns.

[27] The same polysemy characterizes the English verb *tell*.
[28] The analysis of control is not presented in this paper, but is implemented as part of HeGram.

Noun phrases belong to four realization classes: Arg1, Arg2, Arg3, and Arg4. Following are examples of NPs in each of these roles, along with their semantic labels:

(37) (a) [dan $_{Arg1}$] axal [maraq $_{Arg2}$]
Dan ate soup
'Dan ate soup.'

(b) [dan $_{Arg1}$] axal [et ha-maraq $_{Arg2}$]
Dan ate ACC the-soup
'Dan ate the soup.'

(c) [dan $_{Arg1}$] he'exil [et ha-yalda $_{Arg3}$] [maraq $_{Arg2}$]
Dan fed ACC the-girl soup
'Dan fed the girl soup.'

(d) [dan $_{Arg1}$] haya [ha-more šeli $_{Arg4}$]
Dan was the-teacher my
'Dan was my teacher.'

(e) [hi $_{Arg1}$] ʿasta [et dan $_{Arg3}$] [aluf ha-ʿolam $_{Arg4}$]
she made ACC Dan champion the-world
'She made Dan the world champion.'

Accusative case in Hebrew is marked with the case marker *et* only on definite objects.[29] Thus, the indefinite object *maraq* ('*soup*') in (37a) is unmarked, while the definite NP in (37b) is obligatorily preceded by the accusative marker. Accusative case marking also appears on the Affectee/Recipient complement *ha-yalda* ('*the girl*') in (37c), which is associated with Arg3. However, when NPs function as predicates they are never marked with accusative case, regardless of their definiteness status. This is illustrated in the copular construction in (37d) and in (37e), where *ha-more šeli* ('*my teacher*') and *aluf ha-ʿolam* ('*the world champion*') are definite but unmarked for accusative case.

Although all NP complements in (37) are syntactic complements of the verb, their semantic role determines their morphosyntactic behavior. The correlation between the two domains is easily captured in our system, where NP complements are mapped to three distinct semantic roles. The grammar constrains definite Arg2 and Arg3 NP complements to be marked with accusative, and Arg4 NP complements to be unmarked. This generalization applies across the board, regardless of the construction in which NP complements appear.

3.2.5 *Semantic PPs and argument-marking PPs*

Similarly to noun phrases, prepositional phrases also belong to different realization classes. However, not all PPs are equal with respect to their semantic content.

[29] In Hebrew, the accusative case marker is not considered a preposition.

Sag et al. (2003) distinguish between *semantic prepositions* and *argument-marking prepositions*, where the latter do not contribute to the semantics of the clause. Technically, argument-marking prepositions project the index of their NP complements to the PP level. Sag et al. (2003) assume that this information is lexically specified for prepositions, and illustrate this with the lexical entry of *to* (p. 212). In doing so they can account for cases such as *They talk to themselves*, where the complement of the preposition *to* is a reflexive that is bound by the subject.

There are, however, some prepositions that can appear in both guises. Consider, for example, the following pair of sentences:

(38) (a)

[dan $_{Arg1}$]	ṭipel	[ba-tinoq $_{Arg2}$]
Dan	looked-after	in.the-baby

‘Dan looked after the baby.’

(b)

[dan $_{Arg1}$]	sam	[et	ha-sefer $_{Arg2}$]	[ba-argaz $_{Arg7}$]
Dan	put	ACC	the-book	in.the-box

‘Dan put the book in the box.’

The preposition *be* (‘*in*’) (or in its definite form *ba*) is an argument marker in (38a), since it does not contribute any semantic content to the clause (as is evident from the gloss). In (38b), on the other hand, the preposition indicates a location, and can be replaced with other location denoting prepositions such as *ʿal* (‘*on*’), *leyad* (‘*next to*’), or *mul* (‘*opposite*’), resulting, of course, in different meanings.

One possible way to account for this is to posit two different lexical entries for the preposition, one for each function. We, however, propose that the distinction between the two functions of the preposition is not lexically specified; there is only one lexical entry for *be* (‘*in*’), which is underspecified with respect to its semantic status. When heads combine with a PP, the linking of the argument depends on the semantic role of the PP. PPs that are associated with Arg1, Arg2, and Arg3 are considered to be argument markers, and, as such, when they combine with a head, their NP complement (or, more precisely, its index) is considered to be the semantic argument. Semantic PPs, on the other hand, which are associated with Arg4–Arg10, contribute their semantic content to the composition of the semantics of the clause. The technical specifics of this analysis are described in Section 4.4.1 (Figures 12 and 13).

Here, too, the association of complements with finer-grained semantic roles provides a way to express generalizations regarding the behavior of these complements. The proposed analysis attributes the property of being an argument marker or a semantic argument not to the PP itself, but rather to the *relation* between the head and the PP. This eliminates the necessity to posit two distinct lexemes in cases where a preposition serves both functions (with different verbs).

3.2.6 *Complement order*

An additional phenomenon that we find to be governed by semantic constraints is complement order. Unlike English, the order of complements in Modern Hebrew is fairly free. For example, the Arg2 and Arg3 complements of the verb *natan* ('*give*') in (39) can appear in any order.[30]

(39) (a) [dan $_{Arg1}$] natan [le-dana $_{Arg3}$] [et ha-sefer $_{Arg2}$]
dan gave to-dana ACC book

(b) [dan $_{Arg1}$] natan [et ha-sefer $_{Arg2}$] [le-dana $_{Arg3}$]
dan gave ACC book to-dana
'Dan gave the book to Dana.'

A closer examination of the data, however, reveals that not all complement orders are possible. One case that exhibits order constraints is the causative construction, exemplified below:

(40) (a) dana ʿasta [et dan $_{Arg3}$] [adam ṭov $_{Arg4}$]
Dana made ACC Dan man good
'Dana made Dan a good man.'

(b) dana ʿasta [et dan $_{Arg3}$] [me'ušar $_{Arg4}$]
Dana made ACC Dan happy
'Dana made Dan happy.'

(c) *dana ʿasta [adam ṭov $_{Arg4}$] [et dan $_{Arg3}$]
Dana made man good ACC Dan

(d) *dana ʿasta [me'ušar $_{Arg4}$] [et dan $_{Arg3}$]
Dana made happy ACC Dan

While some complement orders may sometimes 'sound better' than others, the sentences in (40c) and (40d), where the predicative Attribute (Arg4) precedes the Affectee object (Arg3), are unequivocally ungrammatical, regardless of the syntactic category of the Attribute.[31]

Consequently, similarly to the phenomena discussed in the previous sections, here too we suggest that the relevant level at which such constraints on complement order apply is the semantic level. More concretely, regardless of the type of syntactic phrase that realizes the predicative complement (Arg4), it cannot precede the Arg3 complement, provided that they are both required. Naturally,

[30] It should be mentioned that this phenomenon poses a challenge to standard HPSG, where complements appear in the COMPS list in a fixed order and are realized accordingly. In an attempt to account for variable complement order, the Matrix (Bender et al. 2002) includes a definition of *basic-head-2nd-comp-phrase*, which realizes the second element in the COMPS list and passes 'upwards' a list of the remaining elements, appended to the initial one. However, as will be presently argued, while this solution may account for some of the cases, it is too general.

[31] As an anonymous reviewer pointed out, this generalization does not take into account possible marked-order constructions such as heavy NP shift. We leave this for future research.

this particular constraint regarding complement ordering is specific to Modern Hebrew. Nevertheless, our approach provides a way of stating semantically based generalizations regarding complement order where applicable.[32]

3.2.7 *Summary*

We reviewed in this section a number of phenomena for which an account that is based on semantics is preferable to one that targets syntactic arguments. We maintain that our proposed list of semantic roles and their realization classes (Table 4), coupled with the lexical representation of verb arguments illustrated above, provides the correct granularity with which argument structure should be expressed. In particular, as we show above, it enables the specification of various generalizations, both semantic and syntactic, that would have been lost with existing approaches.

An additional benefit of our approach to the representation of argument structure is more practical. Computational grammars can be used for parsing, and the analyses provided by our implemented grammar of Hebrew can be used to drive downstream applications which will be able to utilize the semantic representations that the grammar produces. The next section discusses the actual implementation of the grammar that makes such applications possible.

4. Implementation in HPSG

The approach to argument structure that we have presented is implemented in HeGram, a grammar of Modern Hebrew implemented with the LKB (Copestake 1999, 2002b), a grammar development environment. The grammar is based on a starter grammar created with the LinGO Grammar Matrix customization system (Bender, Flickinger & Oepen 2002). Nevertheless, a number of major revisions were made to the 'standard' Matrix-based grammar in order to incorporate this approach:

- The VALENCE feature structure is designed to distinguish between the ten different argument categories.
- Lexical types are cross-classified according to their semantic and syntactic selection.
- Linking between the argument slots of a predicate's key relation and the indices of its arguments is done at the phrasal level.

The interplay of syntax and semantics with respect to argument structure is captured on different levels of generalization: in the lexicon, lexical type hierarchy, and phrasal type hierarchy. This section provides an overview of the essential components of the implemented grammar, from the lexical to the phrasal level.

[32] Note that HPSG is not directional, and constraints can be imposed on multiple levels.

4.1 *The valence complex*

The heart of the analysis lies in the VALENCE feature, which in HeGram is split across XCOMP, which accounts for complex predicates, ten DEP features, each corresponding to a semantic role, and a number of additional features which will be described shortly. The distribution of arguments across several DEP features, in contrast to the standard HPSG use of SUBJ and COMPS, is inspired by Haugereid (2012). However, the specific DEPs, and in particular their association with semantic roles, are novel. An abbreviated description of VALENCE is given in Figure 4.

$$\begin{bmatrix} \text{R-FRAMES} & \textit{link} \\ \text{S-FRAME} & \textit{link} \\ \text{SUBJ-ARG} & \textit{link} \\ \text{DEP1} & \begin{bmatrix} \textit{dependent} & \\ \text{DEP} & \textit{synsem} \\ \text{REAL} & \textit{link} \end{bmatrix} \\ \text{DEP2} & \textit{dependent} \\ \text{DEP3} & \textit{dependent} \\ \ldots & \\ \text{DEP10} & \textit{dependent} \\ \text{XCOMP} & \textit{dependent} \\ \text{PPSORT} & \begin{bmatrix} \text{DEP1-P} & \textit{prep-p} \\ \text{DEP2-P} & \textit{prep-p} \\ \text{DEP3-P} & \textit{prep-p} \\ \ldots & \\ \text{DEP10-P} & \textit{prep-p} \end{bmatrix} \end{bmatrix}$$

Figure 4
The VALENCE feature.

R(EALIZATION)-FRAMES and S(EMANTIC)-FRAME are two features that indicate the types of frames compatible with a particular lexical type. The value of the two features is of type *link*. R-FRAMES, which was introduced in the description of lexical entries in Section 3.2, specifies the realization frames in which the verb can appear. For example, the R-FRAMES value of the first sense of *amar* (‘*tell*’) is *arg12/123/128/1238*. This feature constrains the combination of a head with its subject or complements. S-FRAME specifies the verb’s semantic arguments. Thus, the S-FRAME of the aforementioned sense of *amar* (‘*tell*’) is *arg1238*. The two distinct features are needed in order to account for cases, such as raising or expletives, where a verb combines with a phrase that is not its semantic argument. For example, raised arguments appear in R-FRAMES, but not in S-FRAME. This information is relevant at the phrasal level, where ‘real’ semantic arguments (i.e., those that appear in S-FRAME) are linked to semantic argument slots, while raised arguments are licensed as arguments but not linked.

SUBJ-ARG is used to single out the DEP that functions as the subject. This is explained in more detail in Section 4.4.2. The standard SUBJ and COMPS list members are 'spread' across the ten DEP features. True to the 'one argument per label' principle, the value of each DEP feature is one entity of type *dependent*, rather than a list. The *dependent* complex has two features: DEP, whose value is *synsem*, and REAL, whose value is of type *link*. REAL is used to record the realization of arguments, and is described in more detail in Section 4.4.1.

Finally, the PPSORT complex is used to restrict the types of PPs that can combine with the head for each DEP. The value of each of its DEP-P features is of type *prep-p*, which subsumes atomic preposition types, as well as disjunctive ones.[33]

4.2 *Semantic selection: The* link *hierarchy*

The *link* type hierarchy defines the possible values of R-FRAMES, S-FRAME, SUBJ-ARG, and REAL (Haugereid 2012). It is an elaborate hierarchy, where each semantic frame combination is defined. A highly abbreviated hierarchy illustrating the intricate inheritance relations is given in Figure 5.

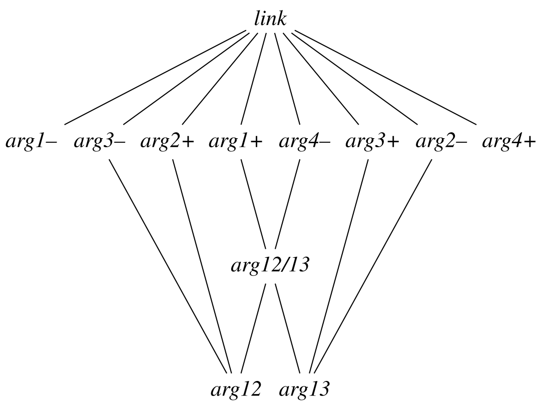

Figure 5
An abbreviated *link* hierarchy.

There are 20 immediate subtypes of *link*, each representing a positive or a negative value of one of the ten *args*. Each 'leaf' in the hierarchy is an (immediate or non-immediate) subtype of one of each pair: *arg1+*, *arg1–*; *arg2+*, *arg2–*; *arg3+*, *arg3–*, etc. For example, *arg12* is a subtype of the positive *arg1+*, *arg2+* and the negative *arg3–* through *arg10–* (not all of which are shown).

[33] Disjunctive preposition types are supertypes that subsume a number of prepositions types. For example, the *m-l_p_rel* value of DEP2-P in Figure 6 is compatible with two preposition types: *m_p_rel* for *me-* ('*from*') and *l_p_rel* for *le-* ('*to*').

Between the general *link* type and the most specific leaf types are disjunctive types. For example, *arg12/13* inherits from the positive *arg1+* and the negative *arg4–* through *arg10–*. It does not inherit from *arg2* and *arg3*, since they are the features where its two disjuncts vary. *Arg12/13*'s immediate subtypes, *arg12* and *arg13*, each representing one of its disjuncts, fully specify their *arg2* and *arg3* values. *Arg12* inherits from the positive *arg2+* and negative *arg3–*, whereas *arg13* inherits from *arg2–* and *arg3+*.

The R-FRAMES feature of predicates determines the combination of a head with its dependents. In order for a predicate to combine with, for example, a DEP2, two conditions must hold. First, its R-FRAMES value must be unifiable with *arg2+*, thus indicating that it indeed selects for an Arg2. Second, the REAL value of its DEP2 must be the negative *arg2–*. Predicates are lexically defined as having negative values in the REAL feature of each of their ten DEPs. Thus, for example, the value of REAL in the *dependent* value of DEP1 is *arg1–*, and similarly in the rest of the DEPs. Once a dependent is realized, its REAL value is set to its respective positive value (*arg1+* in the case of DEP1).

The elaborate structure of the hierarchy is designed in order to support the definition of disjunctive semantic frames such as *arg12/13*. This type is unifiable with either *arg2+* or *arg3+*, and consequently verbs with R-FRAMES of this value are compatible with either Arg2 or Arg3 complements. However, once unification occurs, when the verb combines with the appropriate argument, the R-FRAMES value 'settles' on either *arg12* or *arg13*. This is the key to the mechanism that guides the combination of verbs with their arguments. Disjunctive frames notwithstanding, VPs are required to realize all of the arguments defined in one of the R-FRAMES of their verbal head.[34]

4.3 *Syntactic selection: The* top-deps-lxm *hierarchy*

The specification of the R-FRAMES and S-FRAME values is essentially semantic selection. Syntactic selection involves the specification of constraints on the *synsem* values of the relevant DEP features. Thus, for example, the verb *x̱ašaš* ('*fear*') in (33) (repeated here in (41)) semantically selects an Arg2 and an Arg3.

[34] A fully realized clause is characterized by the unifiability of its head's R-FRAMES value with all of the REAL values of its dependents:

$$\begin{bmatrix} \text{R-FRAMES } \boxed{1} \\ \text{DEP1} \mid \text{REAL } \boxed{1} \\ \text{DEP2} \mid \text{REAL } \boxed{1} \\ \text{DEP3} \mid \text{REAL } \boxed{1} \\ \ldots \end{bmatrix}$$

The realization class associated with Arg2 includes NP, VP, CP, and PP. Of them, *x̱ašaš* (‘*fear*’) syntactically selects VP, PP, and CP. Arg3, on the other hand, can only be realized as a PP, a subset of its realization class {NP, PP}. This information is stated in the *synsem* value of DEP3.[35]

(41) (a) hu x̱ašaš [mi-hefsed $_{Arg2}$] / [le-hefsed $_{Arg2}$] / [lehafsid $_{Arg2}$] /
he feared [from-loss] / [to-loss] / [to.lose] /
[še-hu yafsid $_{Arg2}$]
[that-he will.lose]
‘He feared a loss / a loss / to lose / that he would lose.’

(b) hu x̱ašaš [le-ʿatid-o $_{Arg3}$]
he feared to-future-his
‘He feared for his future.’

(c) *hu x̱ašaš [le-ʿatid-o $_{Arg3}$] [mi-hefsed $_{Arg2}$]
he feared to-future-his from-loss

The VALENCE feature structure of *x̱ašaš* (‘*fear*’) in (41) is presented in Figure 6. Note that disjunctive syntactic selection is represented as a disjunctive HEAD value.[36] *Adp* is shorthand for *adposition*, an order-neutral term for what is usually referred to as ‘preposition’.

R-FRAMES	*arg12/13*
S-FRAME	*arg12/13*
SUBJ-ARG	*arg1*
DEP1	[…HEAD *noun*]
DEP2	[…HEAD *+vpc*]
DEP3	[…HEAD *adp*]
DEP4	*dependent*
…	
PPSORT	[DEP1-P *prep-p*; DEP2-P *m-l_p_rel*; DEP3-P *l_p_rel*; …]

Figure 6
The VALENCE feature of *x̱ašaš* (‘*fear*’).

[35] *Of course x̱ašaš* (‘*fear*’) also selects an Arg1 which can only be realized as an NP, a subset of its realization class {NP, PP}. Note that because Arg1 here must be an NP, the DEP1-P remains unspecified, i.e., *prep-p*.

[36] We adopt the Matrix abbreviations of *head* subtype names: n=noun, v=verb, r=adverb, j=adjective, p=preposition, and c=complementizer. The ‘+’ sign stands for disjunction.

Naturally, information regarding the syntax–semantic correspondences in argument structure is not posited individually for each lexeme. Rather, the lexical type hierarchy in HeGram cross-classifies lexical types according to the different argument slots. Its most general type is *top-deps-lxm*, which dominates all lexical types whose instantiations serve as selecting heads. The immediate subtypes of *top-deps-lxm* are associated with each of the DEP features, and specify a maximal disjoint HEAD value (if applicable). The two highest levels of the hierarchy are illustrated in Figure 7.

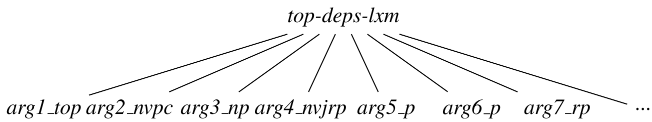

Figure 7
The *top-deps-lxm* hierarchy.

To illustrate the structure of the hierarchy, we will focus on the sub-hierarchy associated with DEP2. Following is the type definition of *arg2_nvpc*, the most general type which dominates types of Arg2-selecting predicates. Note that the type definition below, specified in the TDL formalism (Copestake 2002a), determines the immediate supertype(s) of the particular type (*top-deps-lxm* in this case), as well as type-specific constraints.

(42)
```
arg2_nvpc := top-deps-lxm &
  [ SYNSEM.LOCAL [
        CAT.VAL.DEP2.DEP.LOCAL.CAT.HEAD +nvpc,
        CONT.HOOK.TOPREL.ARG2 semarg ] ].
```

Note the disjoint HEAD value, *nvpc*, which corresponds to the realization class associated with this argument {NP, VP, PP, CP}. The semantic ARG2 specification ensures that the semantic relation denoted by an Arg2-selecting predicate includes an ARG2 slot, regardless of whether this argument is syntactically realized or not.

Below each of the DEP-related general types (e.g., *arg2_nvpc*, *arg3_np*) is a full hierarchy which fleshes out all of the different possible HEAD values. The hierarchy associated with *arg2* is spelled out in Figure 8.

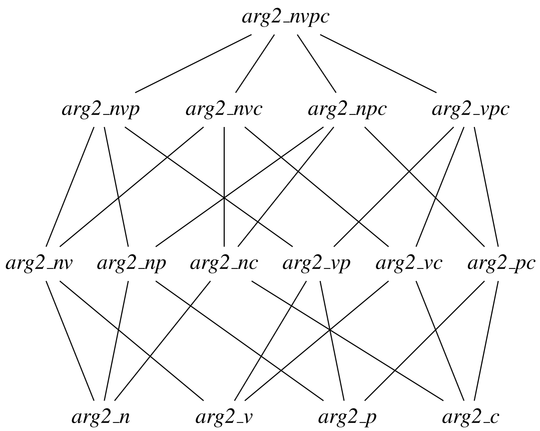

Figure 8
The *arg2* hierarchy.

Maximal verbal types inherit from the appropriate DEP-related types. Following are examples of definitions of two *top-deps-lxm* subtypes, each associated with a different sense of *amar* ('*tell*'). The two definitions correspond to the lexical entries given in (36) (Section 3.2.3). Note that the type associated with the Request sense inherits from *arg2_v*, which defines an atomic HEAD value in its DEP2, while the type associated with the Telling sense inherits from *arg2_nc*, which imposes a disjoint HEAD value (NP or CP). In addition, the Telling sense semantically selects an additional argument (the Topic of Communication, or Arg8), and consequently inherits from the type associated with it, namely *arg8_p*.

(43)
```
;For verbs like amar ('tell') in the Telling sense:
arg12/123/128/1238_nc_p_p :=
  arg1_n & arg2_nc & arg3_p & arg8_p &
  [ SYNSEM.LOCAL.CAT.VAL.R-FRAMES arg12/123/128/1238 ].

;For verbs like amar ('tell') in the Request sense:
arg123_v_p := arg1_n & arg2_v & arg3_p &
  [ SYNSEM.LOCAL.CAT.VAL.R-FRAMES arg123 ].
```

4.4 *Semantic linking at the phrasal level*

HeGram, similarly to grammars implemented within DELPH-IN, produces MRS representations of input sentences, which include specified and underspecified scope information. The key distinction between the MRSs produced by the two approaches is in the argument roles. For example, the semantic relations produced for both *ate* and *fly* in (9a) and (9b), respectively, contain ARG1 and ARG2. HeGram produces more informative representations, where the relations' arguments vary according to their semantic roles: ARG1 and ARG2 (Theme) for *ate*, and ARG1 and ARG6 (Goal) for *flew*.

An additional difference between the two approaches is the level at which argument slots in the semantic relation are linked to the arguments themselves. In standard HPSG this is part of the lexical definition, while in HeGram linking occurs at the phrasal level. This is a direct consequence of the approach which enables the definition of disjunctive argument frames (e.g,. Arg2 can be realized by either NP or CP). Essentially, arguments are linked to argument slots in the head's key relation *as they combine with it*.

4.4.1 *Head–Complement*

A hierarchy of *head–complement* phrase types is defined in order to license the combination of heads with their complements and to define the appropriate semantic links between the semantic relation of the selecting head and its arguments. Semantic linking, as previously mentioned, is standardly assumed to be handled lexically, yet here it is done at the phrasal level.

Consider, for example, a verb such as *gila* ('*discover*'), which may take either an NP or a CP as a complement.

(44)

dan	gila	[et	ha-ocar]	/	[še-ha-kelev	axal	lexem]
Dan	discovered	[ACC	the-treasure]	/	[that-the-dog	ate	bread]

'Dan discovered [the treasure]/[that the dog ate bread].'

In a Matrix-induced grammar, this verb would have had two lexical entries, each associated with a different lexical type.

(45)
```
transitive-lex-item := basic-two-arg-no-hcons &
    [ ARG-ST <
        [ LOCAL.CONT.HOOK.INDEX ref-ind & #ind1 ],
        [ LOCAL.CONT.HOOK.INDEX ref-ind & #ind2 ] >,
      SYNSEM.LKEYS.KEYREL [ ARG1 #ind1,
                            ARG2 #ind2 ] ].
```

(46)
```
clausal-second-arg-trans-lex-item :=
    basic-two-arg &
    [ ARG-ST <
        [ LOCAL.CONT.HOOK.INDEX ref-ind & #ind ],
        [ LOCAL.CONT.HOOK.LTOP #larg ] >,
      SYNSEM [ LKEYS.KEYREL [ ARG1 #ind,
                              ARG2 #harg ],
               LOCAL.CONT.HCONS
                 <! qeq &
                   [ HARG #harg, LARG #larg ] !>
               ] ].
```

Each lexical type handles the linking of the semantics of the complements with the semantic relation of the verb differently, according to the type of the complement.[37] When the verb combines with an NP complement (45), the ARG2 of the semantic relation denoted by the verb is linked with the referential INDEX of the NP. Conversely, when the complement is a CP (46), the content of ARG2 in the semantic relation of the verb is a handle (LTOP). In addition, a *qeq* relation is introduced in HCONS, and it links that handle with the label of the relation denoted by the CP.

The HeGram semantic representation of a clause headed by such a verb is identical, yet the mechanism employed to form it is substantially different. The disjunctive nature of the argument frames of *gila* ('*discover*') is reflected in its lexical definition as an instance of the type *arg12_nc*, which inherits from *arg2_nc* (see Section 4.3). The semantic linking between arguments and argument slots in the semantic relation occurs at the phrasal level.

The basic *head–comp* phrase type is a very general type. Its main function is to percolate VALENCE features which are not relevant to the combination of a head with its complement from the head daughter to the phrase. Its immediate subtype is *basic-head-init-comp-phrase*, which fixes the order to be head-initial. This type, in turn, immediately dominates nine *head–comp* types, each pertaining to one of the nine DEP features: *head-comp2-phrase*, *head-comp3-phrase*, *head-comp4-phrase*, *head-comp5-phrase*, *head-comp6-phrase*, and so on.[38] In what follows we will focus on *head-comp2-phrase* and its subtypes in order to illustrate this aspect of the system. The same mechanism applies to all of its 'sister' phrase types.

As is shown in Figure 9, *head-comp2-phrase* dominates four subtypes, each pertaining to a different type of syntactic phrase within the realization class associated with Arg2 (i.e., NP, CP, VP, and PP). *Head-np-comp2-phrase* is the equivalent of *transitive-lex-item* in terms of the semantic linking it imposes between the NP complement and the semantic relation denoted by the selecting head. Likewise, for CP complements, the semantic links and the *qeq* relation in HCONS, which are defined lexically in the Matrix (see *clausal-second-arg-trans-lex-item* in (46)), are the responsibility of *hd-cp-comp2-ph*. Similar phrase types are defined for the different *head-comp* phrases.[39]

Consider the type definition of *head-comp2-phrase* given in Figure 10. This definition states that a head can combine with a phrase in a *head-comp2-ph* configuration if and only if:

[37] The semantic content is represented using MRS (Copestake et al. 2005). The value of CONT is a feature complex of type *mrs*. Its features include HOOK, RELS, and HCONS. The value of HOOK contains features that specify the parts of the MRS that are visible to semantic functors, among them the INDEX and LTOP, a handle that identifies the label of the (local) topmost predication. This feature complex percolates from the semantic daughter to the mother. RELS is a collection of elementary predications. The RELS of a phrase is the concatenation of the RELS of the daughter. HCONS is a list of handle constraints, which restrict scoping.

[38] The combination of passive verbs with their complements is licensed by a special type *head-comp1-phrase*, which is not included in the hierarchy dominated by *basic-head-init-comp-phrase*.

[39] Note that for space reasons the type names appear in abbreviated forms.

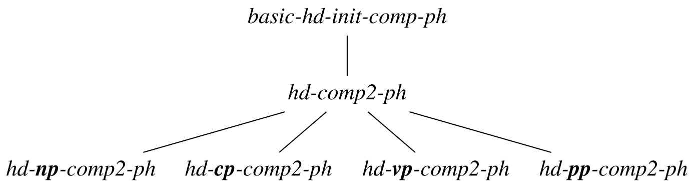

Figure 9
The *hd-comp2-ph* hierarchy.

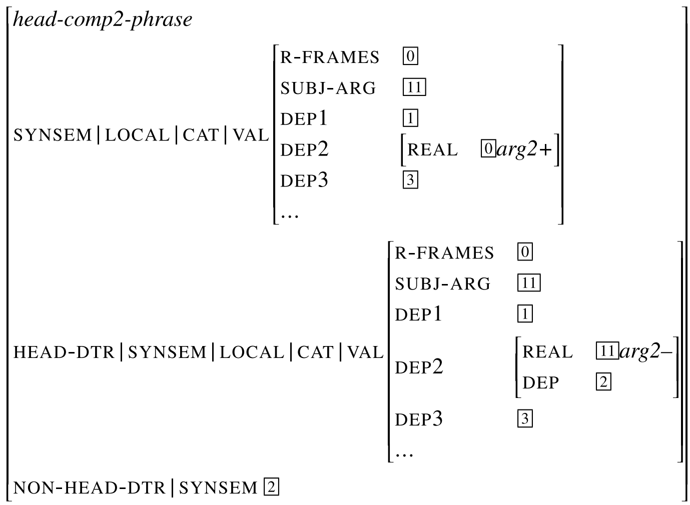

Figure 10
head-comp2-phrase.

- The head subcategorizes for an Arg2.
- The NON-HEAD-DTR is unifiable with the DEP2 requirement of the head.
- The ARG2 requirement of the head has not yet been fulfilled.
- Arg2 is not the head's subject.

Subcategorization constraints are represented in R-FRAMES. The value of the R-FRAMES of the head daughter, tagged [0] in Figure 10, is passed on, along with other VALENCE features, to the mother. Moreover, it is structure-shared with a positive *link*-typed feature. The structure-sharing of the value of R-FRAMES with *arg2+* has two functions. First, it constrains the R-FRAMES value of the head daughter to be compatible with *arg2+*. Thus, for example, a verb whose R-FRAMES value is *arg12/16* is compatible, while *arg16* is not. Second, it can 'force' disjunctive R-FRAMES values to be more restrictive. Thus, a verb with an R-FRAMES of *arg12/16* which combines with an NP complement creates a phrase whose R-FRAMES value is the unification of *arg12/16* with *arg2+*, namely *arg12*. Consequently, once such a verb combines with an NP complement it can no longer combine with a DEP6 argument, since its R-FRAMES value is not compatible with *arg6+*.

In addition, the combination of a head with a complement phrase requires that the non-head daughter be compatible (i.e., unifiable) with the complement requirement specified in the VALENCE feature of the head daughter. The structure sharing between the *synsem* of the non-head daughter and the DEP value of the head daughter's DEP2 value (both tagged [2]) ensures just that.

Unlike in the prevalent approach in HPSG, valence requirements are not canceled off as they are realized. Instead, the grammar keeps track of which arguments have been realized by way of the REAL(IZED) feature, whose value is of type *link*. Lexical items are defined to have negative REAL values in each of their DEPs. Once a head combines with an argument, either a subject or a complement, the value of REAL in its respective DEP is set to a positive value. This is shown in Figure 10, where the value of REAL of the DEP2 of the head daughter is a negative *arg2–*, and in the mother it is a positive *arg2+*. Thus, the specific head–comp rule can only apply once.

The REAL feature is multi-purpose: it allows for free complement order, which is much more difficult to account for with a COMPS list; it enables us to control the complement order when necessary;[40] and it supports disjunctive frames in restricting the realization to only one of the disjuncts (see (33) and Figure 5).

Finally, the structure-sharing of a negative *arg2–* (tagged [11]) with theSUBJ-ARG value of the head daughter determines that DEP2 is not the head's subject. The SUBJ-ARG features plays a key role in the *head-subject-phrase*, described in Section 4.4.2.

The type definition of *head-comp2-phrase* targets the syntactic aspects of the combination of a head with its complement. This is common to all Arg2 complements. The semantic linking between the arguments in the relation denoted by the verb and the complements is subject to phrase-specific constraints, which are defined for each subtype of *head-comp2-phrase*.

Figure 11 illustrates the type definition of *head-cp-comp2-phrase*, a subtype of *head-comp2-phrase*, which accounts for the combination of heads and Arg2 CP complements. As previously mentioned, this phrasal type mimics the semantic linking which is defined lexically in the Matrix (see *clausal-second-arg-trans-lex-item* in (46)). As this phrase type inherits from the more general *head-comp2-ph*, the constraints defined here are only those that are specific to CP complements.

The type-specific information includes all of the information defined in the Matrix for *clausal-second-arg-trans-lex-item* in (46) above. The creation of a *qeq*

[40] The REAL feature is instrumental in defining ordering constraints between complements. It enables the possibility of realizing a complement only if another (specific) complement has been realized, when needed (see Section 3.2.6 about complement order, for elaboration on when this is needed in Hebrew).

$$\begin{bmatrix} \textit{head-cp-comp2-phrase} \\ \text{HEAD-DTR}\,|\,\text{SYNSEM}\,|\,\text{LOCAL}\,|\,\text{CONT}\,|\,\text{HOOK}\,|\,\text{TOPREL}\,|\,\text{ARG2}\ \boxed{2} \\ \text{NON-HEAD-DTR}\,|\,\text{SYNSEM}\,|\,\text{LOCAL} \begin{bmatrix}\text{CAT}\,|\,\text{HEAD}\ \textit{comp} \\ \text{CONT}\,|\,\text{HOOK}\,|\,\text{LTOP}\ \boxed{1}\end{bmatrix} \\ \text{C-CONT}\,|\,\text{HCONS}\left\langle \begin{bmatrix}\textit{qeq} \\ \text{HARG}\ \boxed{2} \\ \text{LARG}\ \boxed{1}\end{bmatrix}\right\rangle \end{bmatrix}$$

Figure 11
head-cp-comp2-phrase.

relation in HCONS is defined in the C-CONT feature (Copestake et al. 2005), whose function is to introduce construction-specific (not lexically based) relations to RELS. Note that the HEAD-DTR's TOPREL feature points to the key relations of the lexical item or the phrase. Its value is set at the lexical level, and it percolates upwards, along with the entire HOOK complex of the head daughter, for all types that inherit from *head-compositional*.[41]

Phrase-type-specific linking constraints are also responsible for capturing the distinction between argument-marking PPs and semantic PPs described in Section 3.2.5. Figure 12 illustrates *head-pp-comp2-phrase*, the type that licenses the combination of a head with an argument-marking PP, while in *head-pp-comp7-phrase* (Figure 13), the complement is a semantic PP. The two phrase types license sentences (38a) and (38b), respectively. The common property of the two is that they both require that the PRED value of the preposition match the restrictions imposed by the head on their respective dependent. This is defined by the structure-sharing tagged $\boxed{1}$. The two types diverge with respect to the semantic contribution of the complement. The ARG2 argument of the head of *head-pp-comp2-phrase* is identified with the ARG2 of the relation denoted by the PP (which in turn is identified with the INDEX of its NP complement). This captures the fact that the head of the PP is only an argument marker. Conversely, with semantic PPs, the ARG7 in the relation denoted by the head is structure-shared with the LTOP (*handle*) of the semantic relation denoted by the PP.

[41] We added the TOPREL feature, which, similarly to the Matrix's LKEYS.KEYREL, is used as a pointer to the main relation, for the purpose of argument linking. However, since in the Matrix argument linking occurs at the lexical level, KEYREL pertains only to lexical *synsem*s. In order to support disjunctive selection (e.g., NP or CP realizing Arg2), HeGram linking occurs as arguments combine with the head. This requires percolating the pointer beyond the lexical level.

$$\begin{bmatrix} \textit{head-pp-comp2-phrase} \\ \text{HEAD-DTR | SYNSEM | LOCAL} \begin{bmatrix}\text{CAT | VAL | PPSORT | DEP2-P } \boxed{1} \\ \text{CONT | HOOK | TOPREL | ARG2 } \boxed{2}\end{bmatrix} \\ \text{NON-HEAD-DTR | SYNSEM | LOCAL} \begin{bmatrix} \text{CAT | HEAD } \textit{adp} \\ \text{CONT | HOOK | TOPREL} \begin{bmatrix}\text{PRED } \boxed{1} \\ \text{ARG2 } \boxed{2}\ \textit{ref-ind}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

Figure 12
head-pp-comp2-phrase.

$$\begin{bmatrix} \textit{head-pp-comp7-phrase} \\ \text{HEAD-DTR | SYNSEM | LOCAL} \begin{bmatrix}\text{CAT | VAL | PPSORT | DEP7-P } \boxed{1} \\ \text{CONT | HOOK | TOPREL | ARG7 } \boxed{7}\end{bmatrix} \\ \text{NON-HEAD-DTR | SYNSEM | LOCAL} \begin{bmatrix} \text{CAT | HEAD } \textit{adp} \\ \text{CONT | HOOK} \begin{bmatrix}\text{TOPREL | PRED } \boxed{1} \\ \text{LTOP } \boxed{7}\ \textit{handle}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

Figure 13
head-pp-comp7-phrase.

4.4.2 *Head–Subject*

The ten DEP features in VALENCE are first and foremost associated with semantic arguments. Consequently, there is no one DEP that represents the subject.

As an illustration of this point consider the following pair:

(47) (a) dan qafac
Dan jumped.3SM
'Dan jumped.'

(b) ha-calaxat nišbera
the-plate.SF broke.3SF
'The plate broke.'

The two verbs are intransitive, yet there is a clear difference between the semantic role of their subject: the subject of the unergative *qafac* ('*jumped*') is an Actor (Arg1), and the subject of the unaccusative *nišbera* ('*broke*') is a Theme (Arg2). The difference between the two verbs is reflected in their VALENCE features. Their respective subject requirements are defined in DEP1 or DEP2, depending on the verb type. Moreover, the subject dependent is identified by the SUBJ-ARG feature, which, too, is part of the VALENCE feature complex. The role of SUBJ-ARG is to determine whether a combination of a dependent is licensed by *head–subj* or *head–comp*.

Similarly to the *head–complement* phrase types, *head–subject* phrases too are defined in a hierarchy, where specific phrase types target specific realizations. As an example, consider the type definition of *head-subj1-phrase*, a subtype of the more general *head-subj-phrase*, given in Figure 14. This phrase type licenses the combination of a VP with its Arg1 subject.

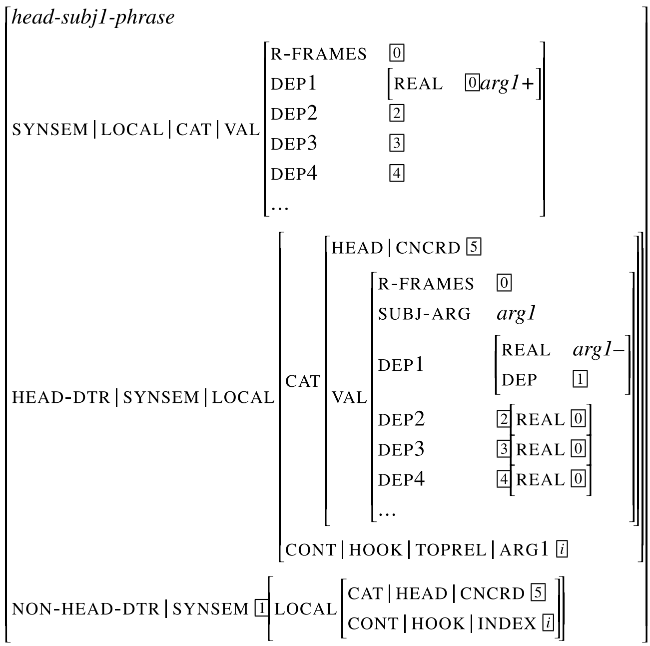

Figure 14
head-subj1-phrase.

The SUBJ-ARG value of the head daughter is restricted to Arg1. Moreover, the structure-sharing of the REAL values of all DEPS except for DEP1 (the subject) determines that the head daughter has realized all of its dependents, except DEP1, whose REAL value is negative.

The SYNSEM of the non-head daughter matches the DEP value of DEP1. Semantic linking is set by structure-sharing the INDEX of the non-head daughter (i.e., the subject) with the value of ARG1 in the TOPREL of the selecting head. In addition, subject–head agreement is reflected in the structure-sharing of the CNCRD feature of the head daughter and the non-head daughter. Finally, the phrase rule sets the REAL feature in DEP1 of the mother to *arg1+*, thus preventing it from combining with another DEP1 again. The combination of Arg2 subjects with VPs is licensed by the 'sister' phrase type, *head-subj2-phrase*, which is identical to the phrase type in Figure 14, except that all specifications of DEP1 and ARG1 are replaced with DEP2 and ARG2.

5. Conclusions

We proposed an analysis that accounts for both the syntactic and the semantic aspects of argument structure. We maintain that argument structure phenomena can be best explained by distinguishing between syntactic selection and semantic selection, while accounting for the correspondences between the two levels.

The schema that we proposed here is fully implemented in HeGram, an HPSG grammar of Modern Hebrew. With this novel infrastructure in place, HeGram covers various constructions including raising, control, unbounded dependencies (wh-questions, topicalization), the Hebrew copula construction, including zero copula, complex predicates, and an inverted (V2) construction.

We are actively working on incorporating verbal multi-word expressions (MWEs) into the grammar (Herzig Sheinfux et al. 2015). We view MWEs as a challenging test case for the innovative architecture that we have implemented. In a way, MWEs constitute an extreme case of multiple subcategorization, so accounting for them is a natural next step in the development of the grammar. We expect that the finer semantic distinctions that our system allows, as well as the explicit differentiation between syntactic constraints and semantic constraints, will prove to be beneficial to the incorporation of MWEs into the grammar.

We regularly run [incr tsdb()] (Oepen 2001) on HeGram, whenever we implement changes in the grammar, and expand our test suite, which currently comprises 483 sentences, out of which 117 are ungrammatical. These sentences, for the most part, target the phenomena that we cover, and a few of them reflect phenomena that we have not covered yet. HeGram is freely available for download.[42]

One argument-structure phenomenon that is not yet accounted for by our grammar is *valence alternation*. Although this study addressed two phenomena that are often included under the term ‘valence alternation’, namely argument optionality and alternate syntactic realization, we view them as separate phenomena. The distinction is based on systematicity. The notion that the syntactic realization of semantic complements is mostly idiosyncratic (within the confines of realization classes) featured in the first example in this paper (i.e., *ask* and *wonder* in (1)). The idiosyncratic and lexeme-specific nature of argument optionality is often illustrated with the verbs *dine*, *devour*, and *eat*, which denote similar activities associated with the consumption of food, yet vary with respect to the optionality of their complement. By ‘valence alternations’ we refer to *systematic* correspondences between different realizations of argument structure for a given lexeme. Common cases are active–passive, causative, and applicative, which generally involve morpholexical operations, but also the English Spray/Load alternation, which is mentioned in Section 2.5. We intend to capture such systematic alternations using valence changing lexical rules, which apply to specific

[42] http://cl.haifa.ac.il/projects/HeGram/index.shtml

lexical types and consequently license additional lexical types with alternate valence specifications.

While the motivation for this representation stems from our attempt to engineer an HPSG grammar of Hebrew, we trust that the schema we propose will be easily adaptable to the specific details of other languages, and be usable in other linguistic theories. The applicability of the schema to other languages is currently being put to the test, as we are beginning to implement an HPSG grammar of Modern Standard Arabic. In this phase of our project we are developing the grammar of the two languages in parallel, sharing fragments of the code for areas where the two languages converge, and defining language-specific types for where they diverge. As the two languages are genetically related, we expect that significant parts of the grammar will be shared.

REFERENCES

Arad Greshler, Tali, Livnat Herzig Sheinfux, Nurit Melnik & Shuly Wintner. 2015. Development of maximally reusable grammars: Parallel development of Hebrew and Arabic grammars. In Müller (ed.), 27–40.

Baker, Carl Leroy. 1968. *Indirect questions in English*. Ph.D. dissertation, University of Illinois at Urbana-Champaign.

Baroni, Marco, Silvia Bernardini, Adriano Ferraresi & Eros Zanchetta. 2009. The WaCky wide web: A collection of very large linguistically processed web-crawled corpora. *Language Resources And Evaluation* 43.3, 209–226.

Bender, Emily M., Dan Flickinger & Stephan Oepen. 2002. The grammar matrix: An open-source starter-kit for the rapid development of cross-linguistically consistent broad-coverage precision grammars. In *Coling-02 Workshop on Grammar Engineering and Evaluation*, 1–7. Morristown, NJ, USA: Association for Computational Linguistics; doi:10.3115/1118783.111878.

Bond, Francis, Stephan Oepen, Eric Nichols, Dan Flickinger, Erik Velldal & Petter Haugereid. 2011. Deep open-source machine translation. *Machine Translation* 25.2, 87–105.

Bonial, Claire, Jena Hwang, Julia Bonn, Kathryn Conger, Olga Babko-Malaya & Martha Palmer. 2012. English PropBank annotation guidelines (Version 3.1) [Annotation guidelines].

Bouma, Gosse, Rob Malouf & Ivan Sag. 2001. Satisfying constraints on extraction and adjunction. *Natural Language & Linguistic Theory* 19, 1–65.

Butt, Miriam. 1995. *The structure of complex predicates in Urdu* (Dissertations in Linguistics Series). Stanford, CA: CSLI Publications.

Chomsky, Noam. 1981. *Lectures on government and binding*. Dordrecht: Foris.

Copestake, Ann. 1999. The (new) LKB system. Technical Report, Stanford University.

Copestake, Ann. 2002a. Definitions of typed feature structures. In Stephan Oepen, Dan Flickinger, Jun-ichi Tsujii & Hans Uszkoreit (eds.), *Collaborative language engineering*, 227–230. Stanford, CA: CSLI Publications.

Copestake, Ann. 2002b. *Implementing typed feature structure grammars*. Stanford, CA: CSLI Publications.

Copestake, Ann. 2009. Slacker semantics: Why superficiality, dependency and avoidance of commitment can be the right way to go. In *The 12th Conference of the European Chapter of the Association for Computational Linguistics*, 1–9. Association for Computational Linguistics.

Copestake, Ann, Dan Flickinger, Carl Pollard & Ivan A. Sag. 2005. Minimal recursion semantics: An introduction. *Research on Language and Computation* 3.2–3, 281–332; http://dx.doi.org/10.1007/s11168-006-6327-9.

Davis, Anthony R. & Jean-Pierre Koenig. 2000. Linking as constraints on word classes in a hierarchical lexicon. *Language* 76.1, 56–91.

Dowty, David. 1991. Thematic proto-roles and argument selection. *Language* 67.3, 547–619.

Emonds, Joseph E. 1991. Subcategorization and syntax-based theta-role assignment. *Natural Language & Linguistic Theory* 9.3, 369–429.

Fillmore, Charles. 1982. Frame semantics. *Linguistics in the morning calm*, 111–137. Seoul: Hanshin Publishing Co.

Fillmore, Charles J. 1968. The case for case. In Bach & Harms (eds.), *Universals in linguistic theory*, 1–88. New York: Holt, Rinehart and Winston.

Fillmore, Charles J. 1971. Some problems for case grammar. *Monograph Series on Languages and Linguistics* 24, 35–56.

Fillmore, Charles J. 2012. Encounters with language. *Computational Linguistics* 38.4, 701–718.

Fillmore, Charles J., Christopher R. Johnson & Miriam R. L. Petruck. 2003. Background to FrameNet. *International Journal of Lexicography* 16.3, 235–250.

Flickinger, Dan. 2000. On building a more efficient grammar by exploiting types. *Natural Language Engineering* 6.1, 15–28; doi:10.1017/S1351324900002370.

Grimshaw, Jane. 1979. Complement selection and the lexicon. *Linguistic Inquiry* 10.2, 279–326.

Gruber, Jeffrey Steven. 1965. *Studies in lexical relations*. Ph.D. dissertation, MIT.

Haugereid, Petter. 2012. A grammar design accommodating packed argument frame information on verbs. *International Journal of Asian Language Processing* 22.3, 87–106.

Herzig Sheinfux, Livnat, Tali Arad Greshler, Nurit Melnik & Shuly Wintner. 2015. Hebrew verbal multi-word expressions. In Müller (ed.), 122–135.

Jackendoff, Ray. 1985. Multiple subcategorization and the ϑ-criterion: The case of climb. *Natural Language & Linguistic Theory* 3.3, 271–295.

Jackendoff, Ray. 1987. The status of thematic relations in linguistic theory. *Linguistic Inquiry* 18.3, 369–411.

Jaworski, Wojciech & Adam Przepiórkowski. 2014a. Semantic roles in grammar engineering. *The 3rd Joint Conference on Lexical and Computational Semantics* (*SEM 2014), 81–86. Association for Computational Linguistics and Dublin City University; http://www.aclweb.org/anthology/S14-1011.

Jaworski, Wojciech & Adam Przepiórkowski. 2014b. Syntactic approximation of semantic roles. *Advances in natural language processing: The 9th International Conference on NLP* (PolTAL 2014), 193–201. Springer International Publishing.

Kaplan, Ronald & Joan Bresnan. 1982. Lexical functional grammar: A formal system for grammatical representation. In Joan Bresnan (ed.), *The mental representation of grammatical relations*, 173–281. Cambridge, MA: MIT Press.

Kingsbury, Paul & Martha Palmer. 2003. Propbank: The next level of treebank. *The Second Workshop on Treebanks and Linguistic Theories*, 105–116. Växjö, Sweden.

Koenig, Jean-Pierre & Anthony R. Davis. 2006. The key to lexical semantic representations. *Journal of Linguistics* 42.1, 71–108.

Levin, Beth. 1993. *English verb classes and alternations: A preliminary investigation*. Chicago: University of Chicago Press.

Müller, Stefan (ed.). 2015. *The 22nd International Conference on Head-driven Phrase Structure Grammar*. Stanford, CA: CSLI Publications.

Oepen, Stephan. 2001. [incr tsdb()] – competence and performance laboratory. User manual. Technical Report, Computational Linguistics, Saarland University, Saarbrücken, Germany.

Palmer, Martha, Dan Gildea & Paul Kingsbury. 2005. The proposition bank: A corpus annotated with semantic roles. *Computational Linguistics* 31.1, 71–106.

Pesetsky, David. 1996. *Zero syntax: Experiencers and cascades*. Chicago, IL/Stanford, CA: University of Chicago Press/CSLI Publications.

Pollard, Carl & Ivan A. Sag. 1994. *Head-driven Phrase Structure Grammar*. Cambridge, MA: MIT Press.

Przepiórkowski, Adam. 1999. On complements and adjuncts in Polish. In Robert D. Borsley & Adam Przepiórkowski (eds.), *Slavic in HPSG*, 183–210. Stanford, CA: CSLI Publications.

Przepiórkowski, Adam, Elżbieta Hajnicz, Agnieszka Patejuk, Marcin Woliński, Filip Skwarski & Marek Świdziński. 2014. Walenty: Towards a comprehensive valence dictionary of Polish. *The 9th International Conference on Language Resources and Evaluation* (LREC-2014), 2785–2792; http://www.lrec-conf.org/proceedings/lrec2014/summaries/279.html.

Sag, Ivan A. & Carl Pollard. 1991. An integrated theory of complement control. *Language* 67, 63–113.

Sag, Ivan A., Thomas Wasow & Emily M. Bender. 2003. *Syntactic theory: A formal introduction*, 2nd edn. Stanford, CA: CSLI Publications.

Schuler, Karin Kipper. 2005. *VerbNet: A broad-coverage, comprehensive verb lexicon*. Ph.D. dissertation, University of Pennsylvania, Philadelphia, PA, USA, AAI3179808.

Sowa, John F. 2000. *Knowledge representation: Logical, philosophical, and computational foundations*. Pacific Grove, CA: Brooks Cole Publishing Co.

Authors' addresses: *(Herzig Sheinfux)*
University of Haifa, Israel
lherzigs@staff.haifa.ac.il

(Melnik)
The Open University, Israel
nuritme@openu.ac.il

(Wintner)
University of Haifa, Israel
shuly@cs.haifa.ac.il