REVISTA SIGNOS. ESTUDIOS DE LINGÜÍSTICA ISSN 0718-0934
© 2017 PUCV, Chile • DOI: 10.4067/S0718-09342017000200241 • 50(94) 241-264

A computational implementation of idiomatic and non-idiomatic constructions[1]

Una implementación computacional de construcciones idiomáticas y no-idiomáticas

Ricardo Mairal Usón
UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA
ESPAÑA
rmairal@flog.uned.es

Recibido: 07-V-2015 / **Aceptado:** 16-XII-2016

Abstract

Departing from a qualitative linguistic analysis based on corpus-attested examples, this paper provides a computational treatment of several construction-types, namely, argument-structure constructions, implicational constructions, and illocutionary constructions. Thus, our aim is to offer a formalized representation of constructions of varied nature and complexity. We do so through a lexico-conceptual knowledge base for natural language processing systems called FunGramKB, whose grammaticon is a computational implementation of the architecture of a usage-based constructionist model of language known as the Lexical Constructional Model.

Key Words: Constructions, Lexical Constructional Model, FunGramKB, Natural Language Processing, COREL.

Resumen

Partiendo de un análisis lingüístico cualitativo basado en ejemplos de corpus, este artículo presenta un tratamiento computacional de varias construcciones, a saber, construcciones argumentales, construcciones implicativas y construcciones ilocutivas. De este modo, nuestro objetivo es ofrecer una representación formal de construcciones gramaticales de diferente naturaleza y complejidad. Para ello empleamos una base de conocimiento léxico-conceptual para sistemas de procesamiento del lenguaje natural denominada FunGramKB, cuyo gramaticón es una implementación computacional de la arquitectura del modelo construccionista basado en el uso llamado Modelo Léxico Construccional.

Palabras Clave: Construcciones, Modelo Léxico Construccional, FunGramKB, Procesamiento del Lenguaje natural, COREL.

INTRODUCTION

The family of Construction Grammars (CxG(s)) comprises a number of compatible approaches, among which we find the following (Hoffmann & Trousdale, 2013): 'Cognitive CxG' (Goldberg, 2006); 'Frame-Semantic CxG' (Boas, 2011); and 'Cognitive Grammar' (Langacker, 1987). In general, even though these accounts differ in their theoretical orientation towards particular issues, most of them share important underlying assumptions (Goldberg, 2013). One point of consensus among the above-listed proposals revolves around the understanding of the notion of linguistic 'construction' as an entrenched form-meaning pairing, which, for any language, is part of a network of form-function correspondences (the so-called 'constructicon') featuring diverse construction-types at different levels of complexity and abstraction. A non-exhaustive inventory includes widely studied argument-structure constructions like the ditransitive (e.g. Scott sent Mary a book), resultative (e.g. The river froze solid), and caused-motion (e.g. The audience laughed the actor off the stage). Above and below the sentence level, conventionalized pieces of language such as 'Hello?! X' (e.g. Hello?! Are you even listening to me?; Google Books American Corpus (GBAC), 2006), 'What's X Doing Y?' (e.g. What's he doing with your money?; GBAC, 2007), 'Could you Y?' (e.g. Could you please hand me that pillow?; GBAC, 2005), 'Let alone' (e.g. We do not hear about her being able to read, let alone speak; GBAC, 2007), to name a few, also count as constructions.

Within the family of CxGs, a specific group of approaches, collectively called 'computational CxGs', includes: 'Embodied CxG' (Bergen & Chang, 2013), 'Sign-Based CxG' (Boas & Sag, 2012), and 'Fluid CxG' (Steels, 2012). Although these approaches are concerned with providing formalized representations of linguistic constructions, most of the work has so far been devoted to argument-structure constructions such as the caused-motion and the ditransitive (e.g. Sag, 2011; van Trijp, 2011; Steels & van Trijp, 2011; Dodge & Petruck, 2014). Similarly, FrameNet, whose aim is "to document the range of semantic and syntactic combinatory possibilities (i.e. valences) of each lexical item" (Ruppenhofer, Ellsworth, Petruck, Johnson & Scheffczyk, 2010: 5), is also building computer-based Constructicons for different languages (Fillmore, 2006, 2008; Fillmore, Lee-Goldman & Rhomieux, 2012; Sato, 2012; Torrent, Meireles, Fernandes, Da Silva & Da Silva, 2014). As will be shown, in the FrametNet English Constructicon, which appears to still be work in progress, the treatment of grammatical constructions is inconsistent in some cases.

In this context, the present paper contributes to this ongoing research program by providing readers with a computational treatment of constructions that exist at different levels of linguistic enquiry. More concretely, our formalized analysis of a representative sample of construction-types departs from a previous qualitative linguistic analysis, which, in line with the usage-based approach, is grounded in a careful examination of corpus-attested examples extracted from systematic searchers within the GBAC at http://googlebooks.byu.edu/x.asp In so doing, we make use of a lexico-conceptual knowledge base for Natural Language Processing (NLP) systems called FunGramKB (Periñán, 2013), whose Grammaticon is a computational implementation of the usage-based constructionist model known as the 'Lexical Constructional Model' (Ruiz de Mendoza, 2013; Ruiz de Mendoza & Galera, 2014). The LCM differs from other constructionist accounts (Goldberg, 1995; Goldberg & Jackendoff, 2004, inter alios), since, in this model, grammatical constructions of varied formal and functional complexity are parsimoniously assigned different places and functions within the same architecture. Such a holistic design is suitable for a computational environment that seeks to include both the propositional and the non-propositional dimensions of meaning. Thus, the LCM distributes heterogeneous constructions across four levels of meaning representation, each of which is computationally implemented in the Grammaticon of FunGramKB: level 1 deals with argument-structure constructions (e.g. He looked for a metal pipe and hammered it flat on one end (GBAC, 2013)), level 2 address implicational constructions (e.g. Don't you honey me!; GBAC, 2007), level 3 focuses on illocutionary constructions (e.g. Can you open the door?; GBAC, 2001), and level 4 is concerned with discourse structure (e.g. Just because I forgive you doesn't mean I forget; GBAC, 2014). In this paper we shall analyze level-1, 2 and 3 constructions from a computational perspective, thus contributing to the formalization of constructions of different formal and functional nature.

With this in mind, the structure of this paper is as follows. Sections 1 describes the nature, linguistic basis and architecture of FunGramKB. For comparative purposes, Section 2, which is devoted to the Grammaticon, first addresses the treatment of grammatical constructions in another knowledge base, i.e. FrameNet. In turn, Sections 2.1, 2.2 and 2.3 respectively focus on argument-structure constructions, implicational constructions and illocutionary constructions. In each of these subsections we provide a constructionist or LCM-based analysis of such construction-types, to then concentrate on a formalized representation of their semantics.

1. FunGramKB

FunGramKB is a user-friendly online environment for the semiautomatic construction of a multipurpose lexico-conceptual knowledge base for NLP systems(see www.fungramkb.com and the publications uploaded therein). First, FunGramKB is semiautomatic since it combines automatic processes (e.g. information in DBPedia is imported onto the Onomasticon) with highly time-consuming tasks such as the manual formalization of knowledge. Second, FunGramKB is multipurpose given its multifunctional and multilingual nature. That is, on the one hand, FunGramKB has been originally designed to be reused in other NLP tasks, and in particular, in those that focus on language understanding (e.g.

machine translation, dialogue-based systems, information retrieval and extraction, etc.). Despite the relative youth of the project, there are already a number of different natural language applications, e.g. Discovering and Extracting Terminology (DEXTER) and Data Mining Encountered (DAMIEN) (see www.fungramkb.com). On the other, FunGramKB faces the challenge of developing a multilingual tool that currently includes several western languages (e.g. Spanish, English, Italian, French, German, etc.). Most of the effort, however, has focused on the development of English and Spanish. Thus, work on the rest of languages is on its early stages.

As far the theoretical framework is concerned, FunGramKB proponents depart from the assumption that although it is easier to build operative NLP systems that do not rely on a given linguistic theory, such systems fail from a semantic point of view (Raskin, 1987). Thus, in order to avoid deceptively intelligent NLP systems and allow natural language understanding instead, it is argued that a linguistic theory must lie at the heart of the computational system. This is also the case of 'Frame Semantics' (Fillmore, 1982) in FrameNet, for example. In FunGramKB this premise translates as follows. The lexical and grammatical knowledge levels, also called Lexicon and Grammaticon, are respectively grounded in two largely compatible linguistic theories. On the one hand, the Lexicon is based on the functional theory of language known as Role and Reference Grammar (RRG) (Van Valin & LaPolla, 1997; Van Valin, 2005). This model, which was not devised for computational linguistics, is nevertheless useful for text meaning representation (Periñán & Arcas, 2014). On the other, the Grammaticon, which also incorporates core aspects of RRG, is inspired in the architecture of the LCM. The explanation behind the introduction of such a constructionist model in FunGramKB has to do with the fact that RRG-based approaches to grammatical constructions are highly underdeveloped (Nolan, 2011, 2014). Consequently, an adequate computational treatment of meaning construction called for the incorporation of a constructionist model which already includes within a single architecture all levels of linguistic enquiry.

The following section describes the architecture of FunGramKB in more detail.

1.1. Architecture

FunGramKB comprises three major knowledge levels that consist of several independent, yet related, modules:

(i) The 'conceptual level' is language-independent and is therefore shared by all the languages currently supported in the knowledge base. This level is made up of three sub-modules, all of which employ the same formal language (i.e. COREL; Periñán & Mairal, 2010) to codify knowledge:

- The Cognicon stores procedural knowledge by means of scripts (e.g. 'going to a restaurant').

- The Onomasticon keeps information about instances of entities and events (e.g. 9/11, Mozart, Eiffel Tower).
- The Ontology, which stores semantic knowledge, takes the form of a hierarchical or IS-A structured catalogue of conceptual units.[2] This component functions as the pivot around which the whole resource revolves. In the Ontology, concepts, to which word senses are connected, are provided with semantic properties in the form of one 'Thematic Frame' (TF) and one 'Meaning Postulate' (MP). It should be noted that both TFs and MPs are treated as conceptual schemata, since they actually employ concepts (instead of words) as the building blocks for the formal description of meaning (Periñán, 2013). A TF states the number and type of participants involved in an event. Partially inspired in Halliday's (1985) and Dixon's (1991) work, participants in FunGramKB, also called 'thematic roles', are expressed through indexed variables (e.g. (x1), (x2), etc.), and their nature may be constrained by concepts that work as 'selectional preferences' (e.g. +CORPUSCULAR_00 in (1)). For example, the TF of the concept +TEAR_01 in (1) is the following:

  (1) TF: (x1)Theme (x2: +CORPUSCULAR_00)Referent = 'An unspecified entity (x1) tears a tridimensional object or (x2)'

In turn, MPs are sets of one or more logically connected predications (e_1, e_2, e_3, etc.) that carry the generic features of concepts. Consider the MP of the concept +TEAR_01, which inherits from its superordinate concept +BREAK_00. The semantics of the lexical units 'tear', 'tear off', 'rip', Spanish *rasgar*, *despedazar*, Italian *strappare*, etc., are related to +TEAR_01, or to put in other words, +TEAR_01 is lexicalized by one sense of each of these lexical items. The formalized representation of +TEAR_01 via the COREL metalanguage is given in (2):

  (2) MP: +(e1: +BREAK_00 (x1)Theme (x2)Referent (f1: +FAST_00)Speed (f2: +HARD_00)Manner) = 'to tear means that an entity (x1) breaks an object (x2) quickly and violently'.

(ii) The 'lexical level' covers what is commonly referred to as linguistic knowledge, which is language-dependent. All lexical units in the Lexica are connected to a concept in the Ontology (e.g. 'rip' in +TEAR_01), from which they get their conceptual definition. This is similar to FrameNet's approach, in which, as is shown below, semantically lexical units evoke frames. Additionally, each lexical unit is individually defined in its corresponding Lexicon. In turn, the role of the Morphicon is to handle cases of inflectional morphology.

(iii) Finally, the 'grammatical level', to which we devote ourselves in Section 2, is also language-specific. This level stores constructional schemata thereby helping RRG to build the semantics-to-syntax algorithm.[3]

2. The Grammaticon: An analysis of constructional schemas across levels of representation

The FunGramKB Grammaticon (Periñán 2013; Periñán & Arcas, 2014; Mairal, 2012, 2015; Van Valin & Mairal, 2014; Luzondo & Ruiz de Mendoza, 2015) is the linguistic module that stores constructional schemata, that is, machine-tractable representations of constructions. It comprises several Constructicons (i.e. L1-Constructicon, L2-Constructicon, L3-Constructicon, and L4-Constructicon), which, as previously noted, are inspired in the four constructional layers of the LCM. The reason for this is the aim to endow FunGramKB with the capacity to deal with both the propositional and non-propositional dimensions of meaning. A similar interest is shared by the FrameNet knowledge base (Baker, 2014), whose English Constructicon already incorporates structures of varying size and complexity such as 'Let alone' (e.g. 'It is difficult enough for an individual to be consistent, let alone a society'), comparison constructions (e.g. 'X is –er than Y'), the 'way' construction (e.g. 'She whistled her way down the lane'), etc. In order to understand the motivation behind the inclusion of specific pieces of language in the FrameNet Construction, we need to briefly describe the lexical resource first.

In FrameNet there are two main units of analysis: i.e. frames and lexical units. A frame is a schematic representations of a situation type (Fillmore, Petruck & Wright, 2003b), which can be defined in terms of participants and their functions. Frames are evoked by a set of lexical units (i.e. words taken in one of their senses). This means that, as much as it is done in FunGramKB, the separate senses of a polysemous word are connected to different semantic frames/concepts. Additionally, FrameNet supplies valence information, which is specified both semantically and syntactically via the following elements: (i)frame elements (i.e. the entities taking part in the situation depicted by a given frame), and (ii) phrase types (e.g. NP, PP, etc.) and their corresponding grammatical functions (Subject, Object, etc.) (Fillmore, Johnson & Petruck, 2003a). Consider the case of the verbal predicate 'tear', which evokes the 'Cause to fragment' frame, i.e. 'An Agent suddenly and often violently separates the Whole_patient into two or more smaller Pieces, resulting in the Whole_patient no longer existing as such'. In this frame definition, Agent, Whole patient, and Pieces are core-frame elements, that is, conceptually prominent components of the frame. In turn, Manner, Time, Purpose, Result, among others, are non-core frame elements, i.e. those that do not uniquely characterize a given frame, and which usually appear across frames. In the lexical entry for 'tear', we are provided with the syntactic patterns in which these frame elements occur, that is, with the valence patterns of this lexical unit. For example, in the corpus attested sentence 'One of the men had torn a paper into little bits', the Noun Phrases 'one of the men' and 'a paper' and the Prepositional Phrase 'into little bits' respectively correspond to the frame elements Agent, Whole patient, and Pieces. Grammatically, these function as Subject (so-called External argument), Dependant and Object, respectively.[4] Similarly, in 'With Black against Karpov it is almost impossible to avoid […] the desire to tear Karpov apart with bare hands', the Whole patient frame element is realized by the NP 'Karpov', which works as an Object. The Result frame element is expressed as an Adverbial Phrase ('apart') which also works as a Dependant. As is obvious from these two instances, classical FrameNet annotations may include realizations of oft-quoted constructions like the resultative. This is possible because the Cause to fragment frame has two frame elements, i.e. Pieces (e.g. 'I smashed the toy boat to flinders') and Result (e.g. 'I broke it open'), which allow the annotation of different resultative ingredients. As Levin (1993: 101)) notes, however, a wide range of verbs can be found in the resultative construction, and thus, "no specific classes of verbs are identified here". These are but a few examples illustrating Levin's statement: 'drink' (e.g. 'John drank himself to death' (GBAC, 2005); 'He drank the glass empty' (GBAC, 2008)), 'cry' (e.g. 'She cried herself to sleep'; Levin 1993), 'kill' (e.g. 'I killed him dead'; (GBAC, 2004)), 'worry' (e.g. 'She's worried sick'; (GBAC, 2009)), 'talk' (e.g. 'He talked himself blue in the face'; (Goldberg, 1995)), etc. In FrameNet, verbal predicates like 'drink', 'cry', or 'talk' are not shown to occur in resultative sentences since the frames which each of these verbs evoke do not display some kind of result element that can then be realized syntactically. By contrast, 'worry' or 'kill' in 'Emotion active' and 'Killing' frames, respectively, do list a non-core Result frame element. This kind of inconsistent annotation of constructions could be easily solved by means of including the resultative construction in the English FrameNet Constructicon, which is thus far not present in such a component. Note that this solution is fully consistent with Torrent et al.'s (2014) analysis of a Brazilian schematic construction, i.e. the Dative with Infinitive construction, which, like the resultative, is not circumscribed to any given verb class. These authors show that, when trying to annotate sentences with verbs that may occur in such a construction (e.g. 'buy', which evokes the 'Commerce_buy' frame, or 'have', which evokes the 'Possession' frame), constituents that can receive a label in one frame, cannot in another frame. This is so because there is no frame element that can be assigned to such a constituent. To account for unlabeled constituents, one could create specific frame elements in each of the frames in which this situation occurs. However, Torrent et al. (2014: 41) argue that:

> "Since several different verbs – evoking several different frames – occur in the Dative with Infinitive construction […] a more general and economic analytical solution – and also a way to avoid inconsistencies of having analogous linguistic material treated different […]– would be including the Dative with Infinitive construction in the […] Constructicon".

In fact, the aim of Torrent et al.'s (2014) work is precisely that of putting forward a set of policies that can help annotators decide what should be accounted for as an instance of a construction, or as a valence pattern of a lexical unit in the lexicographic database. By contrast, border conflicts in the treatment of specific syntactic structures do not pose a problem in FunGramKB in which there is a clearer-cut division between what goes in the Lexicon and what is part of the Grammaticon.

Having pointed out in which way FrameNet and FunGramKB differ in their management of grammatical constructions, the following sections are devoted to the formalization of levels 1, 2 and 3-constructions in order to show how these are handled in an NLP environment.[5]

2.1. The L1-Constructicon

The Grammaticon of FunGramKB is directly connected to the Lexicon, which we briefly presented in Section 1.1. To be more precise, the former is linked to the latter in terms of what has been termed the lexical-grammatical interface (Van Valin & Mairal, 2014). In order to see how lexical units and grammatical constructions interact in FunGramKB, we first devote some space to the Lexicon to then zoom in on the design of the L1-Constructicon. To exemplify this, we shall consider the case of the verbal predicate 'rip' and one of the argument-structure construction in which it participates, i.e. the transitive 'apart reciprocal', as in 'I broke the twig and the branch apart' (Levin, 1993), 'Imperialism rips the governors and the governed apart' (GBAC, 2015). In line with Levin, we argue that this grammatical construction, as well as the 'together reciprocal' construction in 'I creamed the sugar and the butter together' (Levin, 1993), are structural variants of the resultative. In these cases, 'together' and 'apart' can function as resultative adverbs because of their secondary-predication nature. We therefore claim that the 'together/apart reciprocal' construction is part of the family of the resultative (Goldberg & Jackendoff, 2004), whose members express result either in the form of a literal or figurative change of state or as a literal or figurative change of location (see Ruiz de Mendoza & Luzondo, 2016; Luzondo & Ruiz de Mendoza, 2015).

On the one hand, in the Lexicon, the most important component in the case of verbal predicates is called 'core grammar'. Here, lexicographers specify the following elements for the verbal predicate under scrutiny: Aktionsart, variables, macrorole assignment, thematic frame mapping and constructions.[6] These attributes "allow the system to build the basic logical structure of verbs automatically" (Periñán, 2013: 212). By way of illustration, consider the core grammar of English 'rip', a verbal predicate which, as pointed out in Section 1.1, depends on the conceptual information of the ontological concept +TEAR_01:

(3) *Rip* = +TEAR_01

a. Aktionsart: causative accomplishment

b. Variables: **rip'** (*x*, *y*)

c. Macroroles: *x* = Actor; *y* = Undergoer

d. Thematic frame mapping: *x* = Theme; *y* = Referent

e. Constructions in which *rip* participates: transitive resultative (e.g. 'Sue ripped the bag open'; GBAC, 2005); 'apart reciprocal' construction (e.g. 'It tears open the lid of my trunk, seizes my notebooks, rips them apart'; GBAC, 2003); 'way' construction (e.g. 'This opening was no doubt made by the creature in the entryway as it had ripped its way through the exterior of the building to get inside'; GBAC, 2015), etc.

Thus, apart from detailing the verb's Aktionsart adscription ((3a) above), its variables (3b), or how each of these variables maps onto one participant in the thematic frame of the concept to which such a lexical unit is connected (3d), lexicographers must also select, on the basis of corpus evidence, the range of constructions in which the lexical item under scrutiny may appear (3e). In other words, it is by means of the pointers to the repertoire of grammatical constructions with which a given verb may fuse that the projection from syntax to semantics takes place (Periñán, 2013). It should be emphasized that although such pointers are located in the Lexicon, knowledge on constructions is solely stored in the Grammaticon, unlike the case of FrameNet. Additionally, it is worth noting that lexical entries, as well as constructional schemata (see Figure 3), are represented in terms of Attribute-Value Matrices (AVMs). For example, the AVM of 'rip' in Figure 1, which contains the information given in (3), is the following:

Headword	Rip		
Core grammar	CLS	Aktionsart	CACC
	Variables	Type	x
		Role	theme
		Macrorole	actor
		Phrase	NP
		Syntax	argument
		Prefer	-
		Type	y
		Role	referent
		Macrorole	undergoer
		Phrase	NP
		Syntax	argument
		Prefer	+ CORPUSCULAR_00
	L1 - constr	Type	RECI

Figure 1. AVM of the lexical entry for the predicate *rip*.

On the other hand, the interface of the L1-Constructicon is provided in Figure 2:

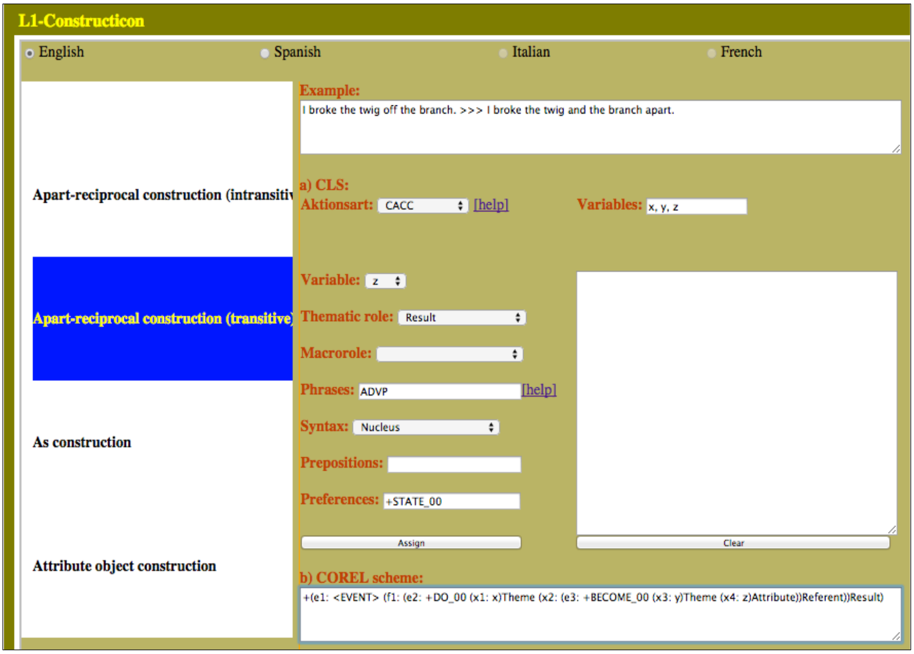

Figure 2. The interface of the L1-Constructicon.

In the interface of the L1-Constructicon, all argument-structure constructions, like the 'apart reciprocal' in Figure 2, are formalized by means of a set of features, which we spell out in (4) below:

(4) a. Aktionsart type: causative accomplishment (CACC)
b. Number of constructional variables: *x*, *y*, *z* (. 'he'; 'the twig and the branch'; 'apart')
c. Variable(s) contributed by the construction: *z* ('apart')
- Thematic role: *z* = result
- Macrorole status: *z* = non applicable
- Morphosyntactic realization: *z* = adverbial phrase (ADVP)
- Syntax (i.e. the status of the new variable in the layered structure of the clause): *z* = nucleus
- Preposition(s): *z* = non applicable
- Selectional preference(s): *z* = +STATE_00
d. COREL schema: +(e1: <EVENT> (f1: (e2: +DO_00 (x1: x)Theme (x2: (e3: +BECOME_00 (x3: y)Theme (x4: z)Attribute))Referent))Result)

As can be seen in (4), FunGramKB constructional schemas are defined by means of two elements, namely, 'descriptors' and 'constraints'. Descriptors include the Aktionsart adscription of the construction, the number of constructional variables involved, the thematic role of the variable(s) contributed by the construction (. *z*, in this case), its macrorole assignment, and the COREL scheme, which captures the cognitive content of constructions. Despite sharing the same formal language, the COREL schema in (4d) differs from the meaning postulate of the concept in (2) in that in the former the variables of the construction are mapped onto one participant role (i.e. (x1: *x*)Theme, (x3: *y*), and (x4: *z*)). For the sake of clarity, the translation of the formalized representation in (4d) is: 'there is an event in which *x* causes *y* to become *z*'. The transitive resultative construction and the 'apart reciprocal' in Figure 2are semantically alike (i.e. they share the same COREL schema). Formally, however, the 'apart reciprocal' construction contains a fixed resultative element (i.e. the adverb *apart*), which distinguishes it from the canonical resultative, whose resultative phrase can take the form of an adjectival phrase (e.g. 'He ate himself sick'; GBAC, 2002) or a metaphorically interpreted prepositional phrase (e.g. 'He ate himself into a coma'; GBAC, 2005). This account is consistent with the claim that both structures, together with the caused-motion construction, the *way* construction, among others, are part of the broader family of the resultative in which constructional members share important functional and formal properties while differing in certain respects (Goldberg & Jackendoff, 2004). In turn, constraints revolve around phrase realizations and selectional preferences. For example, the selectional preference +STATE_00 clearly indicates that *z* changes state as a result of verbal action. By contrast, if we were detailing the constraints of the caused-motion construction (i.e. 'X CAUSES Y TO MOVE TO Z'), for example, the selectional preference +LOCATION_00 would be used instead (e.g. 'He kicked the ball into the net' = 'He caused the object to move to a different location by means of kicking').

All of the information specified thus far is captured in the following AVM:

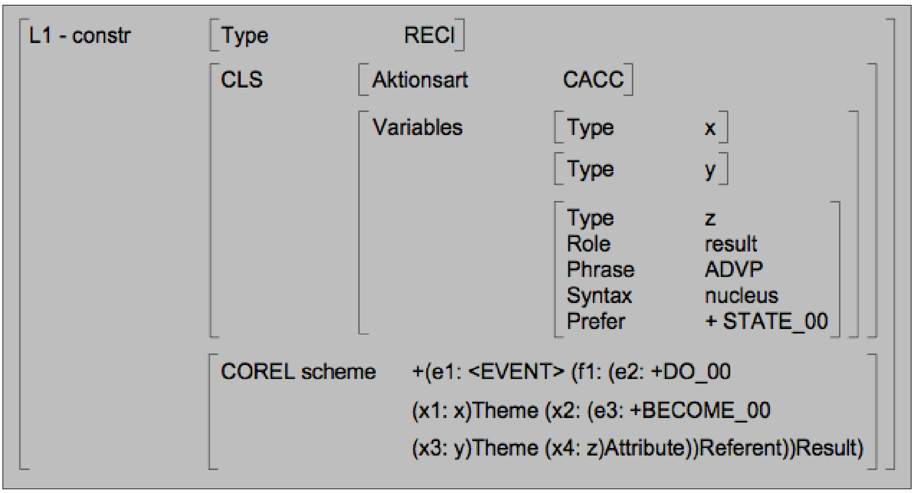

Figure 3. AVM of the 'apart reciprocal' argument-structure construction.

To conclude, consider how lexical-constructional fusion is handled in FunGramKB. The AVM of 'rip' in Figure 1, which contains pointers to the grammatical constructions in which it participates, and the AVM of the 'apart reciprocal' in Figure 3will merge via unification processes following the paradigm of constrain-based grammars (Periñán & Arcas, 2014). In this process, the *x* and *y* variables are inherited from the Lexicon while *z* is the argument contributed by the construction. This allows the machine to successfully interpret an input text like 'It […] rips them apart' as an argument-structure construction.

2.2. The L2-Constructicon

The L2-Constructicon is made up of implicational constructions. One such construction was listed in the Introduction: 'Don't You X Me!', where X is an echo of what the speaker's interlocutor has already said. Note that this construction, like all implicational constructions, captures subjective aspects of meaning, usually in the form of negative emotional reactions like irritation or concern. Two other constructions that convey the speaker's irritation have been discussed in Ruiz de Mendoza and Galera (2014). These are 'Do I Look Like I X?' and 'It Wouldn't Kill You to X'. Like all level 2 constructions, they contain fixed (e.g. 'Look Like') and variable elements (e.g. X). The fixed elements constrain the variable elements from a formal morphosyntactic perspective, but the variable elements are further constrained by non-formal factors too. Since FunGramKB is concerned with conceptualization, we will here explore the meaning part of these two constructions with a view to their computational implementation, to which we devote ourselves at the end of this section.

In the same way as 'Don't You X Me!', the variable part of the 'Do I Look Like I X?' construction is echoic. This is a point that has been noted in Ruiz de Mendoza and Galera (2014), who argue that the combination of an echoic cognitive operation and the clash (or contrast) with the real world situation, if manifest to both speaker and hearer, endows the construction with ironic overtones. For example, the sentence 'Do I look like I want any fucking peaches?' (GBAC, 2000), which realizes this construction, strongly suggests that the speaker is bothered by the hearer's previous assumption, whether explicitly worded or not, that s/he wants peaches. However, there is a difference, in its use of echo, with 'Don't You X Me!' In this latter construction, the echo is necessarily the repetition of what was said before by the speaker's interlocutor (. 'honey' in 'Don't' you honey me!'). In the case of 'Do I Look Like I X?', the variable X can be saturated with any assumption that the speaker believes that the hearer may have in mind that demands some sort of unwanted reaction on the part of the speaker. This feature of the 'Do I Look Like I X?' construction lies at the core of its implicational meaning dimension. More expressions that obey this constraint are the following: 'Do I look like I'm kidding?' (GBAC, 2007), 'Do I look like I understand this?' (GBAC, 2009), 'Do I look like I wanna play poker?' (GBAC, 2006), etc.

There is another important characteristic of this construction that could be added to the initial analysis provided by Ruiz de Mendoza and Galera (2014). This feature has to do with the interesting fact that this construction does not need to repeat back literally what someone else said. The reason for this is to be found in the original inferential process followed by speakers, which has later become consolidated through what cognitive linguists call 'entrenchment' (Langacker, 1999). The variable X is invariably the expression of a state of affairs (i.e. a situation or an event) that the speaker finds unsatisfactory. Even if X is saturated with what in isolation would be axiologically positive, the construction will generally override this meaning component. Contrast:

(5) a. I am the richest man in the world > Do I look like I am the richest man in the world?
b. I am happy > Do I look like I am happy?
c. I feel great today > Do I look like I feel great today?
d. I had a great time with the kids > Do I look like I had a great time with the kids?

It is not difficult to see why this construction can easily perform such a meaning override. If we think of the contexts in which "do I look like" could be reasonably used, the following two elements immediately arise: i) the speaker realizes that the hearer has made an erroneous assumption about his physical appearance and/or attitude (as revealed by bodily posture, facial expression, etc.); ii) because of (i), the speaker realizes that the hearer is not acting in the way the speaker would feel comfortable with. Within this context, the speaker decides on using a communicative strategy that can repair the problems in the hearer's assumptions and associated behavior: iii) the speaker calls the hearer's attention to the true nature of the former's physical appearance and/or attitude (as revealed by bodily posture, facial expression, etc.); iv) in doing (iii) the speaker expects the hearer to change the latter's erroneous assumption in (i) and his associated behavior in (ii). There are several ways in which (iii) could have been done. Here are three of them:

(6) a. I don't want to talk to you, I'm sorry.
b. Why do you think I want to talk to you?
c. Don't think I want to talk to you.

These expressions do not necessarily convey the idea, unlike 'Do I look like I want to talk to you?', that the speaker is bothered by the hearer's erroneous assumption. The formulation with 'do I look like' is constructed in such a way that the speaker shows his surprise that the hearer has made a mistake about what the speaker feels like doing. Since the speaker thinks that there are evident signs of what he wants to do, the implication for the hearer is that the speaker is irritated. Basically, the sentence 'Do I look like I want to talk to you?' can be paraphrased as follows: 'I don't want to talk to you and I think you should have been aware of this by analyzing the situation in which we are both involved in the right way. I am upset that you have not been able to do this and, as a result, you have bothered me by expecting that I will talk to you'.

It goes without saying that expressions realizing the 'Do I Look Like X?' construction can have added illocutionary interpretations. These interpretations are the result of looking at the coded level-2 or implicational meaning from the perspective of interaction rather than just within the personal, subjective dimension discussed here. For example, 'Do I look like I want to talk to you?' could be interpreted as a request for the speaker's interlocutor to leave the speaker alone, or as a reproach, or a complaint, or a warning, or almost any combination of these (Baicchi & Ruiz de Mendoza, 2010; Ruiz de Mendoza, 2015).

The second construction that we want to discuss in this section is 'It Wouldn't Kill You to X' (e.g. 'It wouldn't kill you to help out once in a while'; http://www.ldoceonline.com/dictionary/kill_1), which Ruiz de Mendoza and Galera (2014) have briefly analyzed as a hyperbolic version of 'It Wouldn't Do You Harm to X', which is a form of intensified litotes. Both constructional variants are used to challenge people to take the course of action that the speaker desires or that he thinks it is wise or appropriate for the hearer. This challenge, as the cited authors note, is based on social convention: we are not supposed to ask people to do anything that will be harmful to them. So, clarifying that there is no possible harm involved in the desired course of action becomes the pragmatic equivalent of asking the hearer to take such a course of action. However, there is more to this construction than has been noted in Ruiz de Mendoza and Galera (2014). First, while it is true that the construction expresses a challenge, this is level-2 (or illocutionary) meaning. The construction, at level 2, is used to express the speaker's emotional reaction to the hearer's inaction. The reaction is one of puzzlement: the speaker cannot understand why the hearer has not taken a course of action that will prove beneficial to someone (generally, the speaker or a third party, but it could be the hearer himself). It is from this level-2 meaning that the default illocutionary value (an entreaty or even a challenge) of the construction is obtained. In turn, the variable X is again, as with other constructions, constrained by constructional meaning: X denotes an action/behavior in which the hearer should be actively involved and the action or behavior should be of benefit to someone else, as in 'It wouldn't do you harm/kill you to do the laundry/apologize/follow me on Instagram/help me/be nice/show a little more respect'.

If we saturate X with conceptual material denoting a neutral or non-beneficial action, the construction can either override such a meaning (thus turning it into something beneficial), or, if the result of such an override clashes strongly with what is socio-culturally acceptable, a seriously infelicitous expression may arise:

(7) a. Turn on the computer > It wouldn't do you harm/kill you to turn on the computer.
b. Torture the cat > #It wouldn't do you harm/kill you to torture the cat.

On the basis of the linguistic approach provided thus far, we now turn to a computational treatment of the constructions discussed in this section, i.e. 'Do I Look Like X?' and 'It Wouldn't kill you to X'. For illustrative purposes, Figure 4displays the interface of the L2-Constructicon, showing how the former construction has been formalized:

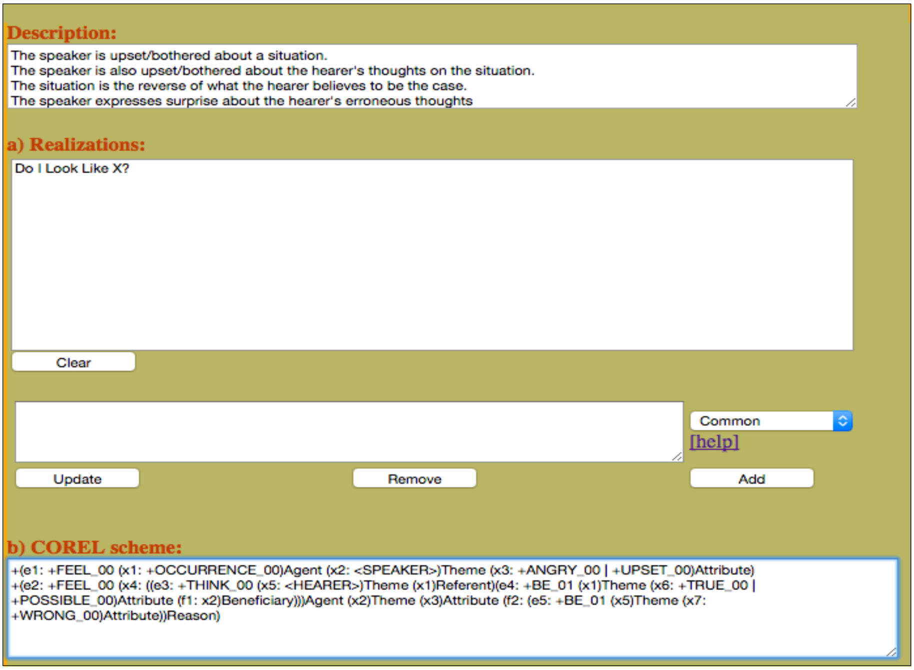

Figure 4. The interface of the L2-Constructicon: The case of *Do I Look Like X?*

As can be seen in Figure 4, the design of the L2 Constructicon differs substantially from that of the L1-Constructicon. This is so because, unlike argument-structure constructions, the only component relevant in the case of both implicational and illocutionary constructions(i.e. L3-Constructicon) is the COREL schema, which carries the meaning of the construction. In turn, the box labeled 'realizations' serves to identify a given input text as an example of a particular construction. This is done through 'pattern-matching'. According to this procedure, if an utterance like 'Do I look like I want to wet-nurse your kids?' (Corpus of Contemporary American English, 2007) were to be processed, the machine would match it with the syntactic pattern 'Do I look like X?', and the relevant meaning implications captured in the COREL schema of the construction would be activated. It is thus essential that the semantic representation of constructional meaning be based on a previous linguistic analysis, which then allows for a fine-grained formal description via COREL. To achieve this, the meaning contributed by the conventionalized construction must be reduced to a set of core predications that can be translated into the COREL notation. For example, departing from the linguistic analysis given above, the semantics of 'Do I look like X?' has been summarized as follows:

(8) a. The speaker is upset/bothered about a situation.
b. The speaker is also upset/bothered about the hearer's thoughts on the situation.
c. The situation is the reverse of what the hearer believes to be the case.
d. The speaker expresses surprise about the hearer's erroneous thoughts on the situation.

These meaning implications are preserved in the COREL schema in Figure 4, which we here repeat for the reader's convenience, together with a translation into natural language:

(9) a. +(e1: +FEEL_00 (x1: +OCCURRENCE_00)Agent (x2: <SPEAKER>)Theme (x3: +ANGRY_00 | +UPSET_00)Attribute) +(e2: +FEEL_00 (x4: ((e3: +THINK_00 (x5: <HEARER>)Theme (x1)Referent)(e4: +BE_01 (x1)Theme (x6: +TRUE_00 | +POSSIBLE_00)Attribute (f1: x2)Beneficiary)))Agent (x2)Theme (x3)Attribute (f2: (e5: +BE_01 (x5)Theme (x7: +WRONG_00)Attribute))Reason)
b. Lit.: There is a situation that makes the speaker feel angry and/or upset. The fact that the hearer thinks that such a situation is true or possible for the speaker makes the speaker feel angry and/or upset because the hearer is wrong.

Similarly, the COREL schema in (10) below captures the cognitive content of the following related constructional realizations: 'It wouldn't kill you to X', 'It wouldn't harm you to X', and 'It wouldn't do you harm to X':

(10) +((e1: n pres +DO_00 (x1: <HEARER>)Theme (x2)Referent (f1)Scene (f2: (e2: +SAY_00 (x3: <SPEAKER>)Theme (x4: (e3: +DO_00 (x1)Theme (x2)Referent))Referent (x1)Goal))Result (f3: (e4: +THINK_00 (x3)Theme (x2)Referent)(e5: pos +BE_01 (x2)Theme (x5: +GOOD_00 | +RIGHT_00)Attribute (f4: x1)Beneficiary))Reason))

The semantic description of these realizations in terms of COREL can be literally paraphrased as follows: 'The hearer is not doing something in a given situation or scene. As a result, the speaker tells the hearer to do (this) something (that he is not doing) because the speaker thinks that this could be beneficial for the hearer'.

Although there are certain functional nuances (e.g. echoing) that cannot be codified through the formal metalanguage, its syntax is flexible enough to capture most of the meaning obtained on inferential grounds.

2.3. The L3-Constructicon

This section concentrates on offers in English. Such an illocutionary category (i.e. ‘offering’)is based on the social convention whereby we are expected to act in ways that are beneficial to other people. This social convention underlies what Leech (1983) called the cost-benefit pragmatic scale within his theory of politeness: polite acts are those in which we maximize benefit and minimize cost to others. But the social convention is more complex. An accurate (and technical) formulation is provided in Ruiz de Mendoza and Baicchi (2007) under the label of Cost-Benefit Cognitive Model. Here, for the sake of simplicity, we sketch out, in a non-technical way, some of its central assumptions:(i) if we identify a state of affairs that is not beneficial to other people, we should alter it in such a way that it becomes beneficial to them; (ii) if we are aware of a state of affairs that could be of benefit to others, we should do our best to bring it about, but (iii) if we suspect that a state of affairs is not beneficial to someone, we should not cause it to come about;(iv) we should make others aware of any course of action that will benefit them; (v) we should make others aware of our willingness to bring about any state of affairs that will benefit them so that they do not need to do it themselves; etc. Each of these assumptions is a stipulation in the cognitive model, which can be exploited in various kinds of illocutionary meaning. For example, stipulation (i) is exploited in expressions like 'Why didn’t you help your mum?' (rebuke), 'Go help your mum, can’t you?' (urging request), 'Your mum needs help' (indirect request). Now, think of offers. These are obviously based on stipulation (v) above, when it builds on (iv). In effect, (iv) works under the assumption that the hearer is not aware that there is a potential benefit for him or her, while in (v) the hearer may or may not be so. If the former case, (v) underlies promises, while in the latter it is used to make offers. There are several conventional ways in which this combination of stipulations (iv) and (v) can be exploited linguistically:

(11) a. Do you want me to do that for you?
b. May I offer you to organize a transfer to the hotel?

Many more such realizations are listed in Del Campo (2013). What is important to note is that they are regularly associated with the specific combination of stipulations pointed out, while highlighting one or another aspect of it: the speaker’s willingness to take the hearer’s wishes into account in (11a), the speaker’s desire to be allowed to assist the hearer in (11b). This highlighting process may have the effect of bringing other stipulations into the interpretive picture. For example, (11a) is focused on the speaker’s realization that there is something that the hearer may need and his subsequent enquiry as to whether the need actually holds. If it does, then the speaker is willing to act in compliance with stipulation (ii).

This brief discussion should be enough to help the reader become aware of the inherent complexity of illocutionary meaning, which is nonetheless susceptible to a computational treatment. To illustrate this, let us consider how the 'offering' scenario has been handled in the L3-construction. This illocutionary scenario, which comprises eleven constructions related through family resemblance (e.g. 'May I Offer You (NP)', 'Is There Anything I Can (VP)?', 'Do You Need Help With NP?', etc.), has been divided into three constructional dimensions, i.e. 'Offering-type 1', 'Offering-type 2' and 'Offering-type 3'. Each of these dimensions comprises several constructions that have been grouped on the basis of their semantic affinity, following the analyses made in Del Campo (2013). Therefore, all the constructions participating in a given dimension share the same COREL schema, as shown in (12)-(14) below:

(12) Offering – Type 1:
a. Semantic description: 'The speaker thinks that he can do something beneficial for the hearer. The hearer may accept the speaker's proposal'.
b. Constructions participanting in this dimension: Do You Want Me (VP)?, I Offer You (NP), Let me (VP) for you, May I Offer You (NP), Will You Let Me(VP)?, Would You Like Me (VP)?
c. COREL schema: +((e1: +THINK_00 (x1: <SPEAKER>)Theme (x2: (e2: pos +DO_00 (x1)Theme (x3)Referent))Referent)(e3: +BE_01 (x3)Theme (x5: +GOOD_00)Attribute (f1: x4)Beneficiary))
+(e4: +SAY_00 (x1)Theme (x6: (e5: +DESIRE_01 (x1)Theme (x2)Referent))Referent (x4: <HEARER>)Goal (f2: (e6: +HELP_00 (x1)Theme (x4)Referent))Purpose)
*(e7: pos +WANT_00 (x4)Theme (x2)Referent)

(13) Offering – Type 2:
a. Semantic description: 'The speaker says that he can do something beneficial for the hearer. The hearer may accept the speaker's proposal'.
b. Constructions participanting in this dimension: Can I Offer You (NP)?, Is There Anything I Can (VP)?, There Must Be Something I Can (VP)
c. COREL schema: +((e1: +SAY_00 (x1: <SPEAKER>)Theme (x2: (e2: pos +DO_00 (x1)Theme (x3)Referent))Referent (x4: <HEARER>)Goal)(e3: +BE_01 (x3)Theme (x5: +GOOD_00)Attribute (f1: x4)Beneficiary (f2: (e4: +HELP_00 (x1)Theme (x4)Referent))Purpose))
*(e4: pos +WANT_00 (x4)Theme (x2)Referent)

(14) Offering – Type 3:
a. Semantic description: The speaker thinks that he can do something beneficial for the hearer. The hearer may accept the speaker's proposal and participate in the bringing about of such a proposal.
b. Constructions participanting in this dimension: Do You Need Help With NP?, May I Help You (VP)?

c. COREL schema: +((e1: +THINK_00 (x1: <SPEAKER>)Theme (x2: (e2: pos +DO_00 (x1)Theme (x3)Referent))Referent)(e3: +BE_01 (x3)Theme (x4: +GOOD_00)Attribute (f1: <HEARER>)Beneficiary)) *(e3: pos +WANT_00 (x5: f1)Theme (x2)Referent (f2: (e4: +HELP_00 (x5)Theme (x2)Referent))Result)

CONCLUSIONS

The aim of this paper was to provide readers with a formalized, machine-tractable treatment of constructions belonging to different levels of linguistic description and explanation. We have thus concentrated on how argument-structure constructions (e.g. the 'apart reciprocal'), implicational constructions (e.g. 'Do I Look Like X?' and 'It Wouldn't kill you to X') and illocutionary constructions(e.g. the 'offering' scenario) are handled in a computational environment like the one provided by FunGramKB, whose grammatical level is a computational implementation of the architecture of the LCM. Therefore, the type of highly formalized analysis presented here contributes to the research program of computational CxGs within the specific context of an NLP system, which has been designed for natural language understanding applications. In addition, our discussion has shown that FunGramKB has benefited from the incorporation of the LCM into its design in two ways. First, construction-types of varied nature and complexity are included within a single architecture. Second, semantic processing is enhanced thanks to the incorporation of the fine-nuanced constructional schemata from the LCM, which go beyond the descriptive and explanatory scope of RRG, the other linguistic model in which FunGramKB is grounded. In turn, FunGramKB has helped to prove the implementability of the LCM within an NLP environment, thereby contributing to its computational adequacy.

REFERENCES

Baicchi, A. & Ruiz de Mendoza, F. (2010). The cognitive grounding of illocutionary constructions within the theoretical perspective of the Lexical Constructional Model. *Textus, English Studies in Italy*, *23*(3), 543-563.

Baker, C. (2014). FrameNet: A knowledge base for natural language processing. In *Proceedings of Frame Semantics in NLP: A workshop in honor of Chuck Fillmore (1929-2014)* (pp. 1-5). Baltimore, Maryland.

Bergen, B. & Chang, N. (2013). Embodied construction grammar. In T. Hoffmann & G. Trousdale (Eds.), *The Oxford handbook of Construction Grammar* (pp. 168-190). Oxford: Oxford University Press.

Boas, H. (2011). Coercion and leaking argument structures in Construction Grammar. *Linguistics*, *49*(6), 1271-1303.

Boas, H. & Sag, I. (Eds.) (2012). *Sign-based construction grammar*. Stanford, CA: CSLI Publications.

Del Campo, N. (2013). *Illocutionary constructions in English: Cognitive motivation and linguistic realization*. Bern: Peter Lang.

Dixon, R. (1991). *A new approach to English grammar on semantic principles*. Oxford: Clarendon Press.

Dodge, E. & Petruck, M. (2014). Representing caused motion in Embodied Construction Grammar. In *Proceedings of the ACL 2014 workshop on semantic parsing* (pp. 39-44). Baltimore: Maryland.

Fillmore, C. (1982). Frame semantics. In The Linguistic Society of Korea (Ed.), *Linguistics in the morning calm* (pp. 11-137). Seoul: Hanshin.

Fillmore, C. (2006). The articulation of lexicon and constructicon. Plenary talk at the Fourth International Construction Grammar Conference (pp. 53-63). University of Tokyo, Tokyo, Japan.

Fillmore, C. (2008). Border conflicts: FrameNet meets Construction Grammar. In *Proceedings of EURALEX13* (pp. 49-68). Barcelona, Universitat Pompeu Fabra.

Fillmore, C., Johnson, C. & Petruck, M (2003a). Background to FrameNet. *International Journal of Lexicography*, *16*(3), 235-250.

Fillmore, C., Petruck, M. & Wright, A. (2003b). FrameNet in action: The case of attaching. *International Journal of Lexicography*, *16*(3), 297-332.

Fillmore, C., Lee-Goldman, R. & Rhomieux, R. (2012). The FrameNet Constructicon. In H. Boas & I. Sag (Eds.), *Sign-Based Construction Grammar* (pp. 309-372). Stanford, CA: CSLI.

Goldberg, A. (1995). *Constructions. A construction grammar approach to argument structure*. Chicago: University of Chicago Press.

Goldberg, A. (2006). *Constructions at work: The nature of generalization in language*. New York: Oxford University Press.

Goldberg, A. (2013). Constructionist approaches. In T. Hoffmann & G. Trousdale (Eds.), *The Oxford handbook of Construction Grammar* (pp. 15-31). Oxford: Oxford University Press.

Goldberg, A. & Jackendoff, R. (2004). The English resultative as a family of constructions. *Language*, *80*(3), 532-568.

Halliday, M. A. K. (1985). *An introduction to functional grammar*. London: Edward Arnold.

Hoffmann, T. & Trousdale, G. (Eds.) (2013). *The Oxford handbook of construction grammar*. Oxford: Oxford University Press.

Langacker, R. (1987). *Foundations of cognitive grammar, Vol. 1: Theoretical prerequisites*. Stanford, CA: Stanford University Press.

Langacker, R. (1999). *Grammar and conceptualization*. Berlin: Mouton de Gruyter.

Leech, G. (1983). *Principles of pragmatics*. London: Longman.

Levin, B. (1993). *English verb classes and alternations. A preliminary investigation*. Chicago & London: University of Chicago Press.

Luzondo, A. & Jiménez, R. (2014). FrameNet and FunGramKB: A comparison of two computational resources for semantic knowledge representation. In B. Nolan & C. Periñán (Eds.), *Language processing and grammars: The role of functionally oriented computational models* (pp. 197-232). Amsterdam: John Benjamins.

Luzondo, A. & Ruiz de Mendoza, F. (2015). Argument structure constructions in a Natural Language Processing environment. *Language Sciences*, *48*(2015), 70-89.

Mairal, R. (2012). La arquitectura de una base de conocimiento léxico conceptual: Implicaciones lingüísticas. In M. Giammatteo, L. Ferrari & H. Albano (Eds.), *Léxico y sintaxis* (pp. 183-210). Mendoza: FFyL, UNCuyo.

Mairal, R. (2015). Constructional meaning representation within a knowledge engineering framework. *Review of Cognitive Linguistics*, *13*(1), 1-27.

Mairal, R. & Periñán, C. (2009). The anatomy of the lexicon component within the framework of a conceptual knowledge base. *Revista Española de Lingüística Aplicada*, 22, 217-244.

Nolan, B. (2011). Meaning construction and grammatical inflection in the layered structure of the Irish word: An RRG account of morphological constructions. In W. Nakamura (Ed.), *New perspectives in Role and Reference Grammar* (pp. 64-101). Newcastle upon Tyne: Cambridge Scholars.

Nolan, B. (2014). Theoretical and computational considerations of linking constructions in Role and Reference Grammar. *Review of Cognitive Linguistics* *12*(2), 410-442.

Ovchinnikova, E., Vieu, L., Oltramari, A., Borgo, S. & Alexandrov, T. (2010). Data-driven and ontological analysis of FrameNet for natural language reasoning. In *Proceedings of the 7th Conference on International Language Resources and Evaluation (LREC'10)* (pp. 3157-3164). Valletta, Malta.

Periñán, C. (2013). Towards a model of constructional meaning for natural language understanding. In B. Nolan & E. Diedrichsen (Eds.), *Linking constructions into Functional Linguistics: The role of constructions in grammar* (pp. 205-230). Amsterdam: John Benjamins.

Periñán, C. & Mairal, R. (2010). La gramática de COREL: Un lenguaje de representación conceptual. *Onomázein*, 21, 11-45.

Periñán, C. & Mairal, R. (2011). The COHERENT methodology in FunGramKB. *Onomázein*, 24, 13-33.

Periñán, C. & Arcas, F. (2014). The implementation of the CLS constructor in ARTEMIS. In B. Nolan & C. Periñán (Eds.), *Language processing and grammars. The role of functionally oriented computational models* (pp. 165-196). Amsterdam: John Benjamins.

Raskin, V. (1987). Linguistics and natural language processing. In S. Nirenburg (Ed.), *Machine translation: Theoretical and methodological issues* (pp. 42-58). Cambridge: Cambridge University Press.

Ruiz de Mendoza, F. (2013). Meaning construction, meaning interpretation and formal expression in the Lexical Constructional Model. In B. Nolan & E. Diedrichsen (Eds.), *Linking constructions into Functional Linguistics: The role of constructions in grammar* (pp. 231-270). Amsterdam: John Benjamins.

Ruiz de Mendoza, F. (2015). Entrenching inferences in implicational and illocutionary constructions. *Journal of Social Sciences*, *11*(3), 258-274.

Ruiz de Mendoza, F. & Baicchi, A. (2007). Illocutionary constructions: Cognitive motivation and linguistic realization. In I. Kecskes & L. Horn (Eds.), *Explorations in pragmatics: Linguistic, cognitive, and intercultural aspects* (pp. 95-128). Berlin & New York: Mouton de Gruyter.

Ruiz de Mendoza, F. & Galera, A. (2014). *Cognitive modeling. A linguistic perspective.* Amsterdam: John Benjamins.

Ruiz de Mendoza, F. & Luzondo, A. (2016). Figurative and non-figurative motion in the expression of result in English. *Language and Cognition*, 8, 32-58.

Ruppenhofer, J., Ellsworth, M., Petruck, M., Johnson, C. R. & Scheffczyk, J. (2010). *FrameNet II: Extended Theory and Practice* [on line]. Retreived from: https://framenet2.icsi.berkeley.edu/docs/r1.5/book.pdf

Sag, I. (2011). Sign-based construction grammar: An informal synopsis. In H. C. Boas & I. Sag (Eds.), *Sign-Based Construction Grammar* (pp. 69-202). Stanford: CSLI.

Sato, H. (2012). A search tool for FrameNet Constructicon. In *Proceedings of the 8th International Conference on Language Resources and Evaluation* (pp. 1655-1658).

Steels, L. (2012). Design methods for Fluid Construction Grammar. In L. Steels (Ed.), *Computational issues in Fluid Construction Grammar* (pp. 3-36). Berlin: Springer.

Steels, L. & van Trijp, R. (2011). How to make Construction Grammars fluid and robust. In L. Steels (Ed.), *Design patterns in Fluid Construction Grammar* (pp. 301-330). Amsterdam: John Benjamins.

Torrent, T., Meireles Lage, L., Fernandes Sampaio, T., da Silva Tavares, T. & da Silva Santos, E. (2014). Revisiting border conflicts between FrameNet and Construction Grammar. *Constructions and Frames, 6*(1), 33-50.

Van Trijp, R. (2011). A design pattern for argument structure constructions. In L. Steels (Ed.), *Design patterns in Fluid Construction Grammar* (pp. 115-145). Amsterdam: John Benjamins.

Van Valin, R. (2005). *The Syntax-semantics-pragmatics interface: An introduction to Role and Reference Grammar*. Cambridge: Cambridge University Press.

Van Valin, R. & LaPolla, R. (1997). *Syntax: Structure, meaning and function*. Cambridge: Cambridge University Press.

Van Valin, R. & Mairal, R. (2014). Interfacing the lexicon and an ontology in a linking system. In M. A. Gómez González, F. Ruiz de Mendoza Ibáñez & F. Gonzálvez-García (Eds.), *Theory and practice in functional-cognitive space* (pp. 205-228). Amsterdam: John Benjamins.

NOTES

[1] Financial support for this research has been provided by the Spanish Ministry of Education and Science, grant FFI2011-29798-C02-01 and FFI2014-53788-C3-1-P. I would also like to thank the three anonymous reviewers for their valuable comments and suggestions. Suffice it say, any mistake is solely mine.

[2] Periñán and Mairal (2011) explain the methodology behind the creation of the Ontology of FunGramKB. For reasons of space, we shall only provide a brief explanation of this core component. The Ontology of FunGramKB is divided into three separate, albeit interrelated, subontologies in which the metaconcepts #ENTITIES, #EVENTS, and #QUALITIES respectively arrange in cognitive dimensions the following parts of speech: (i) nouns, (ii) verbs, and (iii) adjectives. This type of organization stems from the fact that subsumption or IS-A is the only taxonomic relation permitted in the knowledge base.This contrasts with the approach adopted in FrameNet, for example, in which several frame-to-frame relations are posited (see Ruppenhofer et al., 2010). These, however, have been shown to be problematic for NLP as far as reasoning is concerned (see Ovchinnikova, Vieu, Oltramari, Borgo & Alexandrov, 2010). Conceptual, lexical and grammatical information is available through the NLP tool called 'FunGramKB Navigator'. Readers can freely access the Navigators via the following URL: http://www.fungramkb.com/nlp.aspx.

[3] Because RRG is a monostratal theory, syntax and semantics are directly linked without abstract syntactic representations or deep structures. In RRG, there is only one level of representation from the semantic representation of a clause or logical structure to the actual order of constituents. Thus, the theory poses a linking algorithm that contains a number of principles that "illustrate the workings of the syntax-semantics-pragmatics interface" (Van Valin, 2005: 128). One of the distinguishing properties of the RRG linking algorithm is the fact that it is bidirectional, that is, it connects the semantic and syntactic representations as well as the syntactic and semantic representations.

[4] According to Ruppenhofer et al. (2010), constituents occupying core syntactic slots fulfill the functions of Subject and Object. The rest of constituents accompanying a syntactic head are considered Dependents.

[5] See Luzondo and Jiménez (2014) for a detailed comparison between FrameNet and FunGramKB.

[6] The notion of Aktionsart makes reference to the inherent temporal properties of verbs and other predicating elements. For example, 'dry' is a causative accomplishment, that is, it depicts a caused change of state that is neither instantaneous nor atelic. Through the element called 'variables', lexicographers specify the number of prototypical arguments that the verb takes (e.g. 'tear' or 'rip' are bivalent verbs and thus display two arguments, i.e. *x* tears *y*). The number of macroroles that the verbal predicate takes (i.e. Actor, Undergoer) is provided. In RRG, macroroles are groups of semantic roles which function as umbrella notions covering more specific thematic relations (Van Valin, 2005). For example, the Actor (roughly, the logical subject) corresponds to the most agent-like argument, whereas the undergoer (roughly, the logical object) corresponds to the most patient-like participant. In turn, lexicographers link the variables of the predicate to one and only one participant in the TF of the concept to which such lexical unit is connected (e.g. *x* =Theme **tear** *y*=Referent). This is called thematic-frame mapping. For more information on additional elements listed in the Lexicon (e.g. graphical variants, abbreviations, gender, number, pronominzaliation, etc.), which go beyond the scope of the present paper, we refer readers to Mairal and Periñán, 2009).