The Architecture of FunGramKB

Carlos Periñán-Pascual, Francisco Arcas-Túnez
Universidad Católica San Antonio
Campus de los Jerónimos s/n, 30107 Guadalupe (Murcia), Spain
E-mail: jcperinan@pdi.ucam.edu, farcas@pdi.ucam.edu

Abstract

Natural language understanding systems require a knowledge base provided with conceptual representations reflecting the structure of human beings' cognitive system. Although surface semantics can be sufficient in some other systems, the construction of a robust knowledge base guarantees its use in most natural language processing applications, consolidating thus the concept of resource reuse. In this scenario, FunGramKB is presented as a multipurpose knowledge base whose model has been particularly designed for natural language understanding tasks. The theoretical basement of this knowledge engineering project lies in the construction of two complementary types of interlingua: the conceptual logical structure, i.e. a lexically-driven interlingua which can predict linguistic phenomena according to the Role and Reference Grammar syntax-semantics interface, and the COREL scheme, i.e. a concept-oriented interlingua on which our rule-based reasoning engine is able to make inferences effectively. The objective of the paper is to describe the different conceptual, lexical and grammatical modules which make up the architecture of FunGramKB, together with an exploratory outline on how to exploit such a knowledge base within an NLP system.

1. Introduction

FunGramKB Suite[1] is a user-friendly online environment for the semiautomatic construction of a multipurpose lexico-conceptual knowledge base for natural language processing (NLP) systems, and more particularly for natural language understanding. On the one hand, FunGramKB is multipurpose in the sense that it is both multifunctional and multilingual. Thus, FunGramKB has been designed to be potentially reused in many NLP tasks (e.g. information retrieval and extraction, machine translation, dialogue-based systems, etc) and with many natural languages.[2] On the other hand, our knowledge base comprises three major knowledge levels, consisting of several independent but interrelated modules:

Lexical level:

- The Lexicon stores morphosyntactic, pragmatic and collocational information about lexical units.
- The Morphicon helps our system to handle cases of inflectional morphology.

Grammatical level:

- The Grammaticon stores the constructional schemata which help Role and Reference Grammar (RRG) to construct the semantics-to-syntax linking algorithm (Van Valin and LaPolla, 1997; Van Valin, 2005).

Conceptual level:

- The Ontology is presented as a hierarchical catalogue of the concepts that a person has in mind, so here is where semantic knowledge is stored in the form of meaning postulates. The Ontology consists of a general-purpose module (i.e. Core Ontology) and several domain-specific terminological modules (i.e. Satellite Ontologies).
- The Cognicon stores procedural knowledge by means of scripts, i.e. conceptual schemata in which a sequence of stereotypical actions is organised on the basis of temporal continuity, and more particularly on the basis of Allen's temporal model (1983).
- The Onomasticon stores information about instances of entities and events, such as Bill Gates or 9/11. This module stores two different types of schemata (i.e. snapshots and stories), since instances can be portrayed synchronically or diachronically.

In the FunGramKB architecture, every lexical or grammatical module is language-dependent, whereas every conceptual module is shared by all languages. In other words, computational linguists must develop one Lexicon, one Morphicon and one Grammaticon for English, one Lexicon, one Morphicon and one Grammaticon for Spanish and so on, but knowledge engineers build just one Ontology, one Cognicon and one Onomasticon to process any language input conceptually. In this scenario, FunGramKB adopts a conceptualist approach, since the Ontology becomes the pivotal module for the whole architecture.

The rest of this paper is organized as follows. In sections 2 and 3, we explore most of the modules in the FunGramKB conceptual, lexical and grammatical levels. In section 4, we present some of the tools available in FunGramKB Suite. In section 5, we explore the role of FunGramKB when it is integrated into an NLP system. Finally, some conclusions are presented in section 6.

[1] We use the name "FunGramKB Suite" to refer to our knowledge-engineering tool (www.fungramkb.com) and "FunGramKB" to the resulting knowledge base. FunGramKB Suite was developed in C# using the ASP.NET 2.0 platform and a MySQL database.

[2] English and Spanish are fully supported in the current version of FunGramKB Suite, although we have just begun to work with other languages, such as German, French, Italian, Bulgarian and Catalan.

2. The FunGramKB Conceptual Level

The model of "scheme" originated in cognitive psychology, and subsequently implemented in artificial intelligence, is fundamental to the representation of the world knowledge in FunGramKB. In our knowledge base, conceptual schemata are classified according to two parameters: prototypicality and temporality. On the one hand, conceptual representations can store prototypical knowledge (i.e. proto-structures) or can serve to describe instances of entities or events (i.e. bio-structures). For example, the description of the meaning of *song* involves the construction of the proto-structure of its corresponding concept; however, if we want to provide information about the song *Heartbreak Hotel*, then we should do it through a bio-structure. On the other hand, knowledge within conceptual schemata can be presented atemporally (i.e. microstructures) or in a temporal framework (i.e. macrostructures). For example, the biography of Elvis Presley requires a macrostructure; however, a microstructure is sufficient to describe the profession of singer. Therefore, and as shown in Table 1, the convergence of the values of these two parameters results in a typology of four different conceptual schemata which shape the FunGramKB conceptual level.

		TEMPORALITY	
		-	+
PROTOTYPICALITY	+	Proto-microstructure (Meaning postulate)	Proto-macrostructure (Script)
	-	Bio-microstructure (Snapshot)	Bio-macrostructure (Story)

Table 1: Typology of conceptual schemata in FunGramKB.

Tulving (1985) stated that long-term memory components do not work in an isolated way but they interact with each other in order to facilitate information storage and retrieval. Therefore, a key factor for successful reasoning in an NLP system is that all these knowledge schemata must be represented through the same formal language, so that information sharing can take place effectively among all conceptual modules. In FunGramKB, this formal language is COREL (Conceptual Representation Language). To illustrate, (1a) presents the COREL-formatted meaning postulate of +PULL_00, whose natural language equivalent is (1b):[3]

(1a) +((e1: +MOVE_00 (x1: +HUMAN_00 ^ +ANIMAL_00)Agent (x2: +CORPUSCULAR_00)Theme (x3)Location (x4)Origin (x5)Goal (f1: +HAND_00 ^ +MOUTH_00)Instrument (f2: (e2: +SEIZE_00 (x1)Theme (x2)Referent))Condition)(e3: +BE_00 (x1)Theme (x5)Referent))

(1b) A person or animal moves something towards themselves with their hand or mouth, providing that they hold it firmly.

Since the FunGramKB conceptual modules use COREL as the "common language" for schemata representation, natural language understanding systems will only require one "common reasoner". Indeed, we are currently developing an automated cognizer with human-like defeasible reasoning powers which will be able to draw conclusions from information about facts of the real world and knowledge from the repository of FunGramKB meaning postulates, scripts, snapshots and stories. This reasoner is being implemented by using Drools 5.0 platform, which is provided with a forward-chaining inference rules engine whose "native" rule language is powerfully enough so as to preserve the semantic expressivity of COREL. Moreover, Drools supports reasoning over temporal relations between events within an interval-based framework, especially useful for the FunGramKB macrostructures (cf. section 2.2).[4]

2.1 The Core Ontology

The FunGramKB Core Ontology is deemed as an IS-A conceptual hierarchy which allows non-monotonic multiple inheritance. This ontology is both universal and linguistically-motivated.

Firstly, the Core Ontology takes the form of a universal concept taxonomy, where "universal" means that every concept we imagine has, or can have, an appropriate place in the ontology (Corcho, Fernández López and Gómez Pérez, 2001). A universal approach is adopted on the relation between language and conceptualization, where cross-lingual differences in syntactic constructions do not necessarily involve conceptual differences (cf. Jackendoff, 1990).

Secondly, the Core Ontology is linguistically motivated, but not language-dependent. In other words, the Ontology is involved with the semantics of lexical units, but the knowledge stored in the Ontology is not specific to any particular language. In this respect, it is commonly said that the model of the world portrayed in a particular ontology is generally biased by distinctions made in the knowledge engineers' languages (Hovy and Nirenburg,

[3] Periñán-Pascual and Arcas-Túnez (2004) described the formal grammar of well-formed predications in FunGramKB meaning postulates.

[4] In fact, Drools 5.0 implements all the temporal operators defined by Allen's theory (1983).

1992). Consequently, an ontology could be closer to some language communities than to others, finally affecting the ontology design. However, this is not a real problem in FunGramKB, because the structuring of the Ontology is guided by a process of negotiation.

The FunGramKB Core Ontology distinguishes three different conceptual levels, each one of them with concepts of a different type:

(i) Metaconcepts, e.g. #ABSTRACT, #COLLECTION, #EMOTION, #POSSESSION, #TEMPORAL etc, constitute the upper level in the taxonomy. The analysis of the upper level in the main linguistic ontologies—DOLCE (Gangemi et al., 2002), Generalized Upper Model (Bateman, Henschel and Rinaldi, 1995), Mikrokosmos (Mahesh and Nirenburg, 1995), SIMPLE (Lenci et al., 2000), SUMO (Niles and Pease, 2001)—led to a metaconceptual model whose design contributes to the integration and exchange of information with other ontologies, providing thus standardization and uniformity. The result amounts to forty-two metaconcepts distributed in three subontologies: #ENTITY, #EVENT and #QUALITY.

(ii) Basic concepts, e.g. +BOOK_00, +DIRTY_00, +FORGET_00, +HAND_00, +MOVE_00 etc, are used in FunGramKB as defining units which enable the construction of meaning postulates for basic concepts and terminals, as well as taking part as selectional preferences in thematic frames. Instead of adopting a strong approach like that represented by the Natural Semantic Metalanguage (cf. Goddard and Wierzbicka, 2002), which identifies a reduced inventory of semantic primitives that are used to represent meaning, FunGramKB posits an inventory of basic concepts which can be used to define any word in any of the European languages that are claimed to be part of the Ontology. The starting point for the identification of our basic concepts was the defining vocabulary in *Longman Dictionary of Contemporary English* (Procter, 1978), though deep revision was required in order to perform the cognitive mapping into a single inventory of about 1,300 basic concepts.[5]

(iii) Terminals, e.g. $AUCTION_00, $VARNISH_00, $CADAVEROUS_00, $SKYSCRAPER_00, $METEORITE_00 etc, are those concepts which lack definitory potential to take part in the FunGramKB meaning postulates. The hierarchical structuring of the terminal level is very shallow.

[5] This basic level has been tested for validation with the defining vocabulary in the dictionaries of other languages, e.g. *Diccionario para la Enseñanza de la Lengua Española* (VOX-Universidad de Alcalá de Henares, 1995).

2.2 The Cognicon

The FunGramKB script is structured into one or more predications within a linear temporal framework—more particularly, Allen's interval temporal model (1983). This model is based on a representation of time as a partially-ordered graph where nodes represent events and arcs are tagged with one or more relations of temporal ordering. In FunGramKB, every predication included in a script represents an event E which is treated as an interval consisting of a pair of time points (i, t), i.e. the start time-point (i) and the end time-point (t). For example, supposing that an event occurs in the interval E1(i_1, t_1) and another event occurs in the interval E2(i_2, t_2), the interval relation Before(E1, E2), i.e. the event described by the predication e_1 occurs before the event described by the predication e_2, is subject to the constraint $t_1 < i_2$. Moreover, Allen devised an interval-based constraint propagation algorithm which computes all temporal relations taking place when a new event is added to the graph. For instance, and following the previous example, if event E3 is added, and E3 occurs during E2, then the system automatically infers that E1 is before E3. Table 2 presents those interval relations from Allen's model which have been incorporated into the Cognicon.

	Interval relations	Constraints
1.	Before(E1, E2)	($t_1 < i_2$)
2.	Meets(E1, E2)	($i_2 = t_1$)
3.	Overlaps(E1, E2)	($i_1 < i_2$) & ($i_2 < t_1$) & ($t_1 < t_2$)
4.	Starts(E1, E2)	($i_1 = i_2$) & ($t_1 < t_2$)
5.	During(E1, E2)	($i_2 < i_1$) & ($t_1 < t_2$)
6.	Finishes(E1, E2)	($i_2 < i_1$) & ($t_1 = t_2$)
7.	Equals(E1, E2)	($i_1 = i_2$) & ($t_1 = t_2$)

Table 2: Interval relations in the Cognicon.

To illustrate, (2) presents the first nine predications in the classical script @EATING_AT_RESTAURANTS, whose resulting graph is represented in Figure 1.

(2) *(e1: +ENTER_00 (x1: +CUSTOMER_00)Agent (x1)Theme (x2)Location (x3)Origin (x4: +RESTAURANT_00)Goal (f1: (e2: +BE_01 (x1)Theme (x5: +HUNGRY_00)Attribute))Reason)

*(e3: $ACCOMPANY_00 (x6: +WAITER_00)Agent (x6)Theme (x7)Location (x8)Origin (x9: +TABLE_00)Goal)

*(e4: +SIT_00 (x1)Theme (x9)Location)

*(e5: +TAKE_01 (x6)Agent (x10: $MENU_00 | $WINE_LIST_00)Theme (x11)Location (x12)Origin (x9)Goal)

*(e6: +REQUEST_01 (x1)Theme (x13: +FOOD_00 | +BEVERAGE_00)Referent (x6)Goal)

+(e7: +SAY_00 (x6)Theme (x14: (e8: +COOK_00

(x15: $COOK_D_00)Theme (x16: +FOOD_00)Referent))Referent (x15)Goal)

*(e9: +TAKE_01 (x6)Agent (x17: +BEVERAGE_00)Theme (x18)Location (x19: $BAR_00)Origin (x9)Goal)

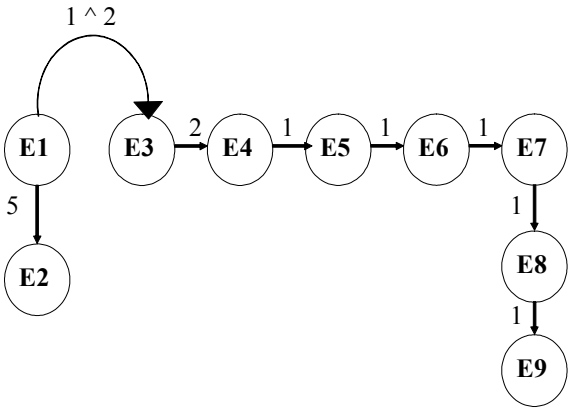

Figure 1: Temporal-knowledge representation in the FunGramKB scripts.

In the FunGramKB scripts, the nodes in the propositional networks can represent either predications or script activators, where the latter include a script identifier and a list of participant-based mappings from the host script to the guest script, as shown in (3).

(3) *(e22: @PAY_CASH_00 (x1: x2, x2: f1))

For instance, within the scenario described by @GOING_SHOPPING_00, we can call another script describing the method of payment, which turns out to be also included in the scripts @GOING_TO_RESTAURANTS_00, @GOING_TO_CINEMAS_00, etc. In this case, @GOING_SHOPPING_00, @GOING_TO_RESTAURANTS_00, @GOING_TO_CINEMAS_00 etc become the host scripts for @PAY_CASH_00, whose role is that of a guest script.

Concerning the full integration of host and guest scripts, and following example (3), participants x1 and x2 in the host script are mapped into x2 and f1 in the guest script respectively. Thus, the FunGramKB script activators explicitly state those participants whose referents in the real world typically coincide.[6] Furthermore, it is possible to call more than one script within the same activator, provided that the disjunctive logical operator is used, as can be seen in (4).

(4) *(e22: @PAY_CASH_00 (x1: x2, x2: f1) ^ @PAY_CARD_00 (x1: x1, x2: x2))

In contrast with the semantic knowledge repository of the Ontology, the possibility of calling a whole script within another script gives us the chance to introduce culturally-biased knowledge in the Cognicon, since every script is assigned a geographical feature determining the continent, country, etc where that knowledge is typically true.[7] Unlike Schank and Abelson's expectation-based model (1977), which deeply influenced the theoretical foundation of the Cognicon, FunGramKB is ready to manage "cultural distinctiveness", which is commonly found in procedural knowledge, e.g. social protocols.

2.3 The Onomasticon

The Onomasticon stores information about named entities and events, i.e. instances of concepts, in the form of bio-structures. To illustrate, we present some of the predications in the snapshot (5a) and the story (6a) assigned to %TAH_MAHAL_00, whose natural language equivalents are presented in (5b) and (6b) respectively:

(5a) +(e1: +BE_02 (x1: %TAH_MAHAL_00)Theme (x2: %INDIA_00)Location)
*(e2: +BE_01 (x1)Theme (x3: +WHITE_00 & $MARBLE_00)Attribute)
*(e3: +COMPRISE_00 (x1)Theme (x4: 1 $DOME_00 & 4 +TOWER_00)Referent)

(6a) +(e1: past +BUILD_00 (x1)Theme (x2: %TAH_MAHAL_00)Referent (f1: 1633)Time)
+(e2: past +BE_00 (x2)Theme (x3: %WORLD_HERITAGE_SITE)Referent (f2: 1983)Time)

(5b) The Tah Mahal is located in India.
Its main material is white marble.
The Tah Mahal has a main dome and four towers.

(6b) The Tah Mahal was built in 1633.
The Tah Mahal became a UNESCO World Heritage site in 1983.

Unlike other FunGramKB modules, the population of the Onomasticon is taking place semi-automatically, by exploiting the DBpedia knowledge base (Bizer et al., 2009). The DBpedia project[8] is intended to extract structured information from Wikipedia, turn this information into a rich knowledge base, which currently describes more than 2.6 million entities, and make this knowledge base accessible on the Web. The population process of the Onomasticon is being performed as follows:

(i) We are manually creating template-based rules which can map the knowledge stored in the DBpedia ontology[9] into COREL-formatted schemata.

[6] Although FunGramKB scripts are clearly interconnected through activators, these macrostructures are not hierarchically organized as in Schank's dynamic memory (1982).

[7] The "default" value of this feature states that a given script is universally applicable.

[8] http://dbpedia.org

[9] The DBpedia ontology, which was manually created from the most commonly-used Wikipedia infoboxes, takes the form of a

(ii) These rules are deployed to FunGramKB Suite, where the mapping occurs automatically.

(iii) Since DBpedia automatically evolves as Wikipedia changes, the Onomasticon will be periodically updated via web service.

3. The FunGramKB Lexical and Grammatical Levels

The FunGramKB lexical model is basically derived from OLIF [10] (McCormick, 2002; McCormick, Lieske and Culum, 2004) and enhanced with EAGLES/ISLE recommendations [11] (Calzolari, Lenci and Zampolli, 2001a, 2001b, 2003; Monachini et al., 2003) with the purpose of designing robust computational lexica. The FunGramKB lexical entries, which can be saved as XML-formatted feature-value data structures, allow the following types of information:[12]

- Basic: headword, index, and language.
- Morphosyntax: graphical variant, abbreviation, phrase constituents, category, number, gender, countability, degree, adjectival position, verb paradigm and constraints, and pronominalization.
- Core Grammar: Aktionsart, lexical template and construction.
- Miscellaneous: dialect, style, domain, example and translation.

Unlike many other NLP lexical databases, the FunGramKB lexical and grammatical levels are grounded in sound linguistic theories, allowing the system to capture syntactic-semantic generalizations which are able to provide both explanations and predictions of language phenomena. Evidently, it is really much easier to build NLP systems when linguistic theories are neglected, but NLP applications which can work perfectly with no foundation in any linguistic theory are deceptively intelligent (Halvorsen, 1988), since they don't allow natural language understanding. In this respect, the linguistic foundation of FunGramKB is inspired on RRG and the Lexical Constructional Model (LCM).

On the one hand, RRG is one of the most relevant functional models on the linguistic scene today. This grammatical model adopts a communication-and-cognition view of language, i.e. morphosyntactic structures and grammatical rules should be explained in relation to their semantic and communicative functions. In RRG, the semantic and the syntactic components are directly mapped in terms of a linking algorithm, which includes a set of rules that account for the syntax-semantics interface. As a result, RRG allows an input text to be represented in terms of a logical structure. For example, the logical structure of the lexical unit *ask for* is (7).

(7) [do' (x, [say' (x, y)])] PURP [do' (y, 0)] CAUSE [BECOME have' (x, z)]

In FunGramKB, the RRG logical structure has been enhanced by a new formalism called "conceptual logical structure" (Periñán-Pascual and Mairal Usón, 2009), so a logical structure such as (7) is now replaced by the representation (8).

(8) [do (x_{Theme}, [+REQUEST_01 (x_{Theme}, y_{Goal})])] PURP [do (y_{Goal}, 0)] CAUSE [BECOME +REQUEST_01 (x_{Theme}, $z_{Referent}$)]

The main benefits of CLSs can be summarized as follows:

(i) CLSs are real language-independent representations, since they are made of concepts and not words. One of the consequences of this interlingual approach is that redundancy is minimized while informativeness is maximized.

(ii) The inferential power of the reasoning engine is more robust if predictions are based on cognitive expectations. In order to perform some reasoning with the input, the CLS should be transduced into a COREL representation, so that it can be enriched by the knowledge in meaning postulates, scripts, snapshots and stories.

Therefore, CLSs serve to build a bridge between the FunGramKB conceptual level and the particular idiosyncrasies coded in a given linguistic expression. For instance, the sentence *Betty asked Bill for an apple* has the CLS (9), which can be mapped into the COREL representation (10).

(9) $<_{IF}$ DECL $<_{TNS}$ PAST <[do (%BETTY_00_{Theme}, [+REQUEST_01 (%BETTY_00_{Theme}, %BILL_00_{Goal})])] PURP [do (%BILL_00_{Goal}, 0)] CAUSE [BECOME +REQUEST_01 (%BETTY_00_{Theme}, +APPLE_$00_{Referent}$)]>>>

(10) +(e1: past +REQUEST_01 (x1: %BETTY_00)Theme (x2: +APPLE_00)Referent (x3: %BILL_00)Goal)

In this CLS-COREL mapping process, the grammatical operators, the FunGramKB concepts and their thematic roles are the only CLS elements taken into account.

shallow IS-A hierarchy of 170 classes, containing 720 properties.

[10] OLIF (Open Lexicon Interchange Format) is an XML-compliant standard for lexical/terminological data encoding.

[11] ISLE (International Standards for Language Engineering), which is an extension of EAGLES work, supports R&D on human-language technology issues.

[12] Mairal Usón and Periñán-Pascual (2009) presented the anatomy of the FunGramKB Lexicon by describing the different types of features which form part of a predicate's lexical entry.

On the other hand, the LCM (Ruiz de Mendoza and Mairal, 2008; Mairal and Ruiz de Mendoza, 2009), which is grounded in the RRG framework, goes beyond the core grammar. The LCM incorporates meaning dimensions that have a long tradition in pragmatics and discourse analysis. Thus, the LCM recognizes the following four levels of constructional meaning:

(i) Level 1, or *argumental layer*, accounts for the core grammatical properties of lexical items.

(ii) Level 2, or *implicational layer*, is concerned with the inferred meaning related to low-level situational cognitive models (or specific scenarios), which give rise to meaning implications of the kind that has been traditionally handled as part of pragmatics through implicature theory.

(iii) Level 3, or *illocutionary layer*, deals with traditional illocutionary force, which is considered a matter of high-level situational models (or generic scenarios).

(iv) Level 4, or *discourse layer*, addresses the discourse aspects, with particular emphasis on cohesion and coherence phenomena.

In the Grammaticon, each one of these constructional levels is computationally implemented into a Constructicon. Thus, the CLS (9) is automatically generated by means of the Core Grammar of the verb together with the grammatical information in the L1-Constructicon. Furthermore, CLSs can be incrementally expanded by each type of Constructicon. For instance, an L3-CLS is that logical structure which has been enriched by the implicational and illocutionary levels of constructional meaning.

Currently most NLP lexical databases—e.g. SIMPLE (Lenci et al., 2000) or EuroWordNet (Vossen, 1998), among many others—adopt a relational approach to represent lexical meanings, since it is easier to state associations among lexical units in the way of meaning relations than describing the conceptual content of lexical units formally. However, although large-scale development of deep-semantic resources requires a lot of time, effort and expertise, not only is the expressive power of conceptual meanings much more robust, but the management of their knowledge also becomes more efficient (cf. Periñán-Pascual and Arcas-Túnez, 2007).

4. Tools in FunGramKB Suite

FunGramKB Suite is provided with a set of user-friendly tools to browse, check and edit the knowledge base. To illustrate, some of these tools are briefly described:

(i) Conceptual, lexical and grammatical modules can be browsed via a GUI, displaying specific feature-value information about their elements.

(ii) When building conceptual knowledge in the form of meaning postulates, scripts, snapshots or stories, a syntactic-semantic validator is triggered, so that consistent well-formed constructs can be stored.

(iii) In order to help knowledge engineers to determine the granularity of meaning postulates, a checklist suggests the semantic components which could become relevant on the basis of the conceptual dimension to which the concept belongs.

5. Integrating FunGramKB into an NLP System

One of the first attempts to integrate FunGramKB into an NLP system is aimed at improving the performance of UniArab, an Arabic-to-English machine translator (Nolan and Salem, 2009; Salem, Hensman and Nolan, 2008a, 2008b; Salem and Nolan, 2009a, 2009b). The advantage of UniArab lies in the deployment of an interlingua architecture which uses a robust functional linguistic model founded on RRG in the machine translation kernel. On the one hand, UniArab is built upon an interlingua machine translation architecture, which is more flexible and scalable for multilingual generation. On the other hand, one of the primary strengths of UniArab is the accurate representation of the RRG logical structure of an Arabic sentence. To illustrate, the logical structure (11) is built from the Arabic sentence (12), whose translation into English is sentence (13).

(11) <TNS:PAST[do'(Khalid,[read'(Khalid,(book)])]>

(12) قرأ خالد الكتاب

(13) Khalid read the book.

Currently, UniArab covers a representative broad selection of words and can translate simple sentences including intransitive, transitive and ditransitive clauses, as well as copular-like nominative clauses. Concerning the evaluation of UniArab, this system clearly outperforms existing machine translators in the processing of simple sentences, suggesting that RRG is a promising candidate for interlingua-based machine translation. In fact, Salem and Nolan (2009b) demonstrated that UniArab provides more accurate and grammatically-correct translations than statistical machine translators such as Google (2009) and Microsoft (2009).

However, the model of UniArab devised by Brian Nolan and his research team fails to provide an adequate treatment of the semantics of lexical units. For instance, UniArab avoids the problem of word sense disambiguation by adopting a naive one-word-one-sense approach to lexical polysemy. To overcome this problem, among many others, the UniArab lexical database is replaced by FunGramKB, where lexical entries are more

informative and meaning capabilities are deeper. Thus, at the end of the syntax-semantics processing, the enhanced version of UniArab generates a syntactic representation of the input where lemmas have been replaced by conceptual tags. In the case of sentence (12), the output would be the parenthetical representation (14), whose concepts are also provided with lexico-conceptual information represented as feature-value matrices.

(14) S(NP(n(%KHALID_00)), VP(v(+READ_00), NP(det(the), n(+BOOK_00))))

The RRG logical structure (11) is then developed out of the phrasal structure (14), but now taking the form of the CLS (15).

(15) $<_{IF}$ DECL $<_{TNS}$ PAST < do ($KHALID_00_{Theme}$, [+READ_00 ($KHALID_00_{Theme}$, +$BOOK_00_{Referent}$)] & INGR +READ_00 (+$BOOK_00_{Referent}$)>>>

The shift from the standard RRG model of logical structure to the CLS approach opens a new avenue for UniArab to cope with complex multilingual input.

6. Conclusions

This paper addresses the three levels of knowledge which shape FunGramKB, a multipurpose knowledge base for NLP systems. We highlight two main contributions of our project in comparison with other similar knowledge bases. On the one hand, the FunGramKB conceptual level enables the full integration of semantic, procedural and episodic knowledge by sharing both the knowledge representation language and the reasoning engine. As a result, expectations on the occurrence of typical events in a given situation are based on COREL schemata, a concept-oriented interlingua whose inferential power is greater than the traditional approach to lexical semantics. On the other hand, the FunGramKB lexico-grammatical levels are grounded in a solid linguistic theory in order to capture syntactic-semantic generalizations which can manage and interpret data. In this respect, both the RRG and the LCM frameworks inspired the construction of the CLS, a lexically-driven interlingua through which the system is able to predict a wide range of linguistic phenomena (e.g. passivization) in the language generation process. Whereas the CLS serves as the pivot language between the input text and the COREL representation, the latter serves as the pivot language between the CLS and the automated reasoner. Consequently, the primary goal of our project is the development of an NLP knowledge base sufficiently robust to help language engineers to design intelligent natural language understanding systems.

7. Acknowledgements

Financial support for this research has been provided by the DGI, Spanish Ministry of Education and Science, grant FFI2008-05035-C02-01/FILO. The research has been co-financed through FEDER funds.

8. References

Allen, J. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11), pp. 832--843.

Bateman, JA., Henschel, R., Rinaldi, F. (1995). The Generalized Upper Model 2.0. Technical report. IPSI/GMD, Darmstadt.

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S. (2009). DBpedia: a crystallization point for the Web of Data. *Journal of Web Semantics: Science, Services and Agents on the World Wide Web*, 7, pp. 154--165.

Calzolari, N., Bertagna, F., Lenci, A., Monachini, M. (Eds.) (2003). Standards and Best Practice for Multilingual Computational Lexicons and MILE. Deliverable D2.2-D3.2. ISLE Computational Lexicon Working Group.

Calzolari, N., Lenci, A., Zampolli, A. (2001a). The EAGLES/ISLE computational lexicon working group for multilingual computational lexicons. In *Proceedings of the First International Workshop on Multimedia Annotation*. Tokyo.

Calzolari, N., Lenci, A., Zampolli, A. (2001b). International standards for multilingual resource sharing: the ISLE Computational Lexicon Working Group. In *Proceedings of the ACL 2001 Workshop on Human Language Technology and Knowledge Management*. Morristown, NJ: Association for Computational Linguistics, pp. 71--78.

Corcho, O., Fernández López, M., Gómez Pérez, A. (2001). Technical Roadmap v.1.0. IST-OntoWeb Project, Universidad Politécnica de Madrid.

Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., Schneider, L. (2002). Sweetening ontologies with DOLCE. In *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web*. London: Springer, pp. 166--181.

Goddard, C., Wierzbicka, A. (Eds.) (2002). *Meaning and Universal Grammar*. Amsterdam: John Benjamins.

Google (2009). Google Translator. http://translate.google.com.

Hovy, E., Nirenburg, S. (1992). Approximating an interlingua in a principled way. In *The DARPA Speech and Natural Language Workshop*, New York.

Jackendoff, R. (1990). *Semantic Structures*. Cambridge, Mass.: MIT Press.

Lenci, A., Bel, N., Busa, F., Calzolari, N., Gola, E., Monachini, M., Ogonowski, A., Peters, I., Peters, W., Ruimy, N., Villegas, M., Zampolli, A. (2000). SIMPLE: a general framework for the development of multilingual lexicons. *International Journal of Lexicography*, 13(4), pp. 249--263.

Mahesh, K., Nirenburg, S. (1995). Semantic classification for practical natural language processing. In *6th ASIS SIG/CR Classification Research Workshop: An Interdisciplinary Meeting*. Chicago: ASIS, pp. 79--94.

Mairal Usón, R., Periñán-Pascual, C. (2009). The anatomy of the lexicon component within the

framework of a conceptual knowledge base. *Revista Española de Lingüística Aplicada*, 22, pp. 217--244.

Mairal Usón, R., Ruiz de Mendoza, F. (2009). Levels of description and explanation in meaning construction. In C.S. Butler & J. Martín Arista (Eds.), *Deconstructing Constructions*. Amsterdam/Philadelphia: John Benjamins, pp. 153--198.

McCormick, S. (2002). The Structure and Content of the Body of an OLIF v.2.0/2.1. The OLIF2 Consortium.

McCormick, S., Lieske, C., Culum, A. (2004). OLIF v.2: A Flexible Language Data Standard. The OLIF2 Consortium.

Microsoft (2009). Microsoft Translator. htttp://www.windowslivetranslator.com/Default.aspx.

Monachini, M., Bertagna, F., Calzolari, N., Underwood, N., Navarretta, C. (2003). Towards a Standard for the Creation of Lexica. ELRA European Language Resources Association.

Niles, I., Pease, A. (2001). Origins of the Standard Upper Merged Ontology: a proposal for the IEEE Standard Upper Ontology. In *Working Notes of the IJCAI-2001 Workshop on the IEEE Standard Upper Ontology*. Seattle.

Nolan, B., Salem, Y. (2009). UniArab: an RRG Arabic-to-English machine translation software. In *Proceedings of the Role and Reference Grammar International Conference*. Berkeley.

Periñán-Pascual, C., Arcas-Túnez, F. (2004). Meaning postulates in a lexico-conceptual knowledge base. In *Proceedings of the 15th International Workshop on Databases and Expert Systems Applications*. Los Alamitos, CA: IEEE, pp. 38--42.

Periñán-Pascual, C., Arcas-Túnez, F. (2007). Deep semantics in an NLP knowledge base. In *Proceedings of the 12th Conference of the Spanish Association for Artificial Intelligence*, Salamanca, pp. 279--288.

Periñán-Pascual, C., Mairal Usón, R. (2009). Bringing Role and Reference Grammar to natural language understanding. *Procesamiento del Lenguaje Natural*, 43, pp. 265--273.

Procter, P. (Ed.) (1978). *Longman Dictionary of Contemporary English*. Harlow: Longman.

Ruiz de Mendoza Ibáñez, F., Mairal, R. (2008). Levels of description and constraining factors in meaning construction: an introduction to the Lexical Constructional Model. *Folia Linguistica*, 42(2), pp. 355--400.

Salem, Y., Hensman, A., Nolan, B. (2008a). Implementing Arabic-to-English machine translation using the Role and Reference Grammar linguistic model. In *Proceedings of the 8th Annual International Conference on Information Technology and Telecommunication,* Galway.

Salem, Y., Hensman, A., Nolan, B. (2008b). Towards Arabic to English machine translation. *ITB Journal*, 17, pp. 20--31.

Salem, Y., Nolan, B. (2009a). Designing an XML lexicon architecture for Arabic machine translation based on Role and Reference Grammar. In *Proceedings of the 2nd International Conference on Arabic Language Resources and Tools,* Cairo.

Salem, Y., Nolan, B. (2009b). UniArab: a universal machine translator system for Arabic Based on Role and Reference Grammar. In *Proceedings of the 31st Annual Meeting of the Linguistics Association of Germany*.

Schank, R.C. (1982). *Dynamic Memory*. New York: Cambridge University Press.

Schank, R., Abelson, R.P. (1977). *Scripts, Plans, Goals and Understanding*. Hillsdale: Lawrence Erlbaum.

Tulving, E. (1985). How many memory systems are there? *American Psychologist*, 40, pp. 385--398.

Van Valin, R.D. Jr. (2005). *The Syntax-Semantics-Pragmatics Interface: An Introduction to Role and Reference Grammar*. Cambridge: Cambridge University Press.

Van Valin, R.D. Jr., LaPolla, R. (1997). *Syntax, Structure, Meaning and Function*. Cambridge: Cambridge University Press.

Vossen, P. (1998). Introduction to EuroWordNet. *Computers and the Humanities*, 32(2-3), pp. 73--89.

VOX-Universidad de Alcalá de Henares. (1995). *Diccionario para la Enseñanza de la Lengua Española*. Barcelona: Bibliograf.