Lexfom: a lexical functions ontology model

Alexsandro Fonseca
Université du Québec à Montréal
201 Président Kennedy Avenue
H2X 3Y7, Montreal, Canada
affonseca@gmail.com

Fatiha Sadat
Université du Québec à Montréal
201 Président Kennedy Avenue
H2X 3Y7, Montreal, Canada
sadat.fatiha@uqam.ca

François Lareau
Université de Montréal
C.P. 6128 succursale Centre-Ville
Montreal, Canada
francois.lareau@umontreal.ca

Abstract

A lexical function represents a type of relation that exists between lexical units (words or expressions) in any language. For example, the antonymy is a type of relation that is represented by the lexical function *Anti*: *Anti*(*big*) = *small*. Those relations include both paradigmatic relations, i.e. vertical relations, such as synonymy, antonymy and meronymy and syntagmatic relations, i.e. horizontal relations, such as objective qualification (*legitimate demand*), subjective qualification (*fruitful analysis*), positive evaluation (*good review*) and support verbs (*pay a visit*, *subject to an interrogation*). In this paper, we present the *Lexical Functions Ontology Model* (lexfom) to represent lexical functions and the relation among lexical units. Lexfom is divided in four modules: *lexical function representation* (lfrep), *lexical function family* (lffam), *lexical function semantic perspective* (lfsem) and *lexical function relations* (lfrel). Moreover, we show how it combines to *Lexical Model for Ontologies* (lemon), for the transformation of lexical networks into the semantic web formats. So far, we have implemented 100 simple and 500 complex lexical functions, and encoded about 8,000 syntagmatic and 46,000 paradigmatic relations, for the French language.

Keywords: lexical functions, lexical ontology, lexical network, collocations

1 Introduction

We present in this paper the Lexical Functions Ontology Model (lexfom), a model for the representation of lexical functions (Mel'čuk, 1998) of the Meaning-Text Theory (MTT) (Mel'čuk, 1997).

A lexical ontology uses the semantic web formalism (RDF/OWL languages) to represent different aspects of the lexicon, such as meaning, morphology, part of speech, as well as the relation among lexical units, such as syntactic, semantic and pragmatic relations.

We show in this paper how our ontology can be used to represent relations among lexical units in lexical networks. This is an important aspect since most of the existing lexical networks do not implement syntagmatic information (Schwab et al., 2007) provided by some Lexical Functions (LFs). Moreover, we show how this model can be used to represent collocations in a lexical network since the relation among lexical units in a collocation is a syntagmatic relation (Mel'čuk 1998).

We do not intend to recreate lexical representations already realized by previous works, such as lemon (McCrae et al., 2012), LexInfo (Buitelaar, 2009) or LMF (Francopoulo, 2007). Our proposal is to use, whenever possible, the lexical information already implemented by those models, such as the classes "LexicalEntry" and "LexicalSense" in the lemon model, and create the necessary classes for the implementation of lexical functions information.

The present paper is organized as follows: in the Section 2, we report state-of-the-art related to the problem. Section 3 presents our proposed scheme for an ontology to represent LFs. In Section 4 we give a summary of the lexical functions and lexical relations encoded within our model. Finally, Section 5 summarizes our work and gives future perspective.

2 Foundations and Related Work

We present in this section the theoretical information about LFs and related work, as follows: In Section 2.1, we give the definition of collocation adopted in this paper. Section 2.2 explains LFs and gives some examples. In Section 2.3, we discuss the *French Lexical Network* (in French, *Réseau Lexical du Français*), which is based on LFs. In Section 2.4, we present a semantic classification for LFs. Finally, in Section 2.5, we discuss the lemon model and how we intend to combine it with our model to represent sense relations in a lexical network.

2.1 Definition of collocation

Before giving the definition of collocation, we present an example to show how frequent collocations are and the importance of treating them in computer applications. This example was taken from Mel'čuk (2004):

> Government troops have spread a DRAGNET across the country in a SEARCH for three heavily ARMED guerrillas. The FARC has claimed RESPONSIBILITY for the ATTACK launched Tuesday in which four ROCKETS were fired at an ARMY camp.

In this example, each underlined expression is a collocation. The capitalized word is the base or keyword of the collocation and the non capitalized word is the collocate. Note that collocates have a more idiomatic than prototypical meaning in each collocation and ignoring them can cause problems in machine translation, information retrieval and text generation applications.

A phrase is unrestrictedly constructed when the rules used in its construction are not obligatory. For example, instead of saying "pay for a lunch" we could say "pay for a meal". In contrast, an expression such as "pay attention" is fixed. We cannot say "pay care", even if it is grammatically correct. Therefore, "pay attention" is a phraseme, since it is not unrestrictedly constructed.

A phase is regularly constructed when its words are combined according to general rules of a grammar and its sense can be derived exclusively from sense of its constituent words. The phrase "red house" is regularly constructed because it follows the rules of the English grammar and its sense can be obtained from its constituent words. On the other hand, the expression "red neck" is not regularly constructed: it follows the rules of the English grammar. However, its sense cannot be completely derived from its constituent words.

A collocation is a kind of phraseme, as defined by Mel'čuk (1998). There are two types of phasemes: pragmatic phrasemes or pragmatemes and semantic phrasemes, as defined by Morgan (1978). The pragmatemes are defined as:

- Expressions whose signified and signifier (Saussure, 1983) are not unrestrictedly constructed, even if they are regularly constructed. For example: "all you can eat", "see you later";
- Expressions whose signified only is not unrestrictedly constructed. For example: greetings, technical expressions and phrases like "it is forbidden to smoke".

In semantic phrasemes, the signified is free (it is unrestrictedly constructed, although it may not be regularly constructed) and its signifier is not free. There are three types of semantic phrasemes:

- Idioms: the sense of the expression is larger than the sense of its constituent words, which are not included in the sense of the expression. Examples: "to kick the bucket", "to spill the beans";

- Quasi-phrasemes or quasi idioms: the signified of the expression includes the signified of its constituent words. However, it also contains a signified that goes beyond the signified of each isolated word. Example: "start a family", "bed and breakfast";
- Collocations: The signified of the expression includes the signified of one of its constituent words (w_1), which is freely chosen, and another word or expression, (w_2), which is chosen contingent to (w_1). There are different types of collocations (Manning and Schütze, 1999): light verbs constructions (*to pay attention*, *to make a decision*), phrasal verbs (to *take out*, *to give up*), etc.

According to Polguère (2000), a collocation is a semi-idiomatic expression having the form L_1+L_2, where one of the components, the collocate (L_2) is chosen to express a specific sense in a specific syntactic role contingent to the selection of the other component, the base or keyword (L_1). The selection of collocate depends strongly on the lexeme chosen as keyword (Heid and Raab 1989).

2.2 Lexical Functions

Bolshakov and Gelbukh (1998) defined a lexical function (LF) as a formalism for the description and use of combinatorial properties of individual lexemes. A more technical definition, given by Mel'čuk (1998), says that a "*Lexical Function f* is a function that associates with a given lexical unit *L*, which is the argument, or keyword, of *f*, a set {L_i} of (more or less) synonymous lexical expressions – the *value* of *f* – that are selected contingent on *L* to manifest the meaning corresponding to *f*:

$$f(L) = \{L_i\}$$

The LFs considered in this paper are the standard ones, differentiated from the non-standard by the fact that the former can be coupled with a higher number of possible keywords and value elements (Mel'čuk 1998). For example, the LF *Bon*, which represents the sense "subjective qualifier", can be coupled with many keywords (e.g. cut_N, $struggle_N$, *proposal*, *service*, *place* and many others) to give different values : Bon(cut_N) = {*neatly*, *cleanly*}; Bon($struggle_N$) = *heroic*; Bon(*proposal*) = *tempting*; Bon(*service*) = *first-class*; Bon(*place*) = *prominent*; (Mel'čuk 1998). On the other hand, the sense "*additionné de…*" (with the addition of…) is a non-standard LF in French, because it can only be coupled with a few number of keywords (*café*; *fraises*; *thé*), to create the expressions: *café crème*, *fraises à la crème* (and not **café à la crème*, **fraises crème*); *café au lait*; *café arrosé*; *café noir*; *thé nature*; etc (Mel'čuk 1992). About 70 simple standard LFs have been identified (Kolesnikova, 2011).

LFs can be classified as paradigmatic or syntagmatic, according to the kind of lexical relation they model. The paradigmatic LFs model the vertical, "in absence" or "in substitution" relation among lexical units (Saussure, 1983). For example, antonymy, Anti(*big*) = *small*; synonymy, Syn(*car*) = *automobile*; hyponymy, Hypo(*feline*) = {*cat*, *tiger*, *lion*, etc.}. Syntagmatic LFs model the horizontal, "in presence" or "in composition" relations among lexical units (Saussure, 1983). For example: magnification, Magn(*committed*) = *deeply*; confirmation, Ver(*argument*) = *valid*; laudatory, Bon(*advice*) = {*helpful*, *valuable*}.

Another important concept is that of semantic actant (Sem-actant) (Mel'čuk, 2004). In logic, a predicate is a falsifiable assertion. Each predicate has one or more arguments. For example, in the assertion "Rome is the capital of Italy", we can define the predicate 'capital' having two arguments, 'Rome' and 'Italy': *capital*(*Italy*, *Rome*).

In linguistics, the predicate is called "predicative sense" and the arguments are its "semantic actants". Each LF represents a different predicative sense and the semantic actants are represented by subscripts. For example, the LF *S* (actantial noun) gives the equivalent noun of the value to which it is applied. S_1 gives the first actant (the one who executes the action), S_2 gives the second actant (the object of the action) and S_3 gives the third actant (the recipient of the action): S_1(*to teach*) = *teacher*; S_2(*to teach*) = {*subject*; *matter*}; S_3(*to teach*) = {*pupil*; *student*}. Other subscripts give circumstantial information. For example: S_{loc} – local of the action/event; S_{instr} – instrument used; etc.

LFs can be classified according to their semantic or syntactic behaviour. For example, in (Mel'čuk, 1998) we find the following classification:

- Semantic derivatives: S_1(*to teach*) = *teacher*; S_3(*to teach*) = *pupil*; S_{loc}(*to fight*) = *battlefield*; S_{instr}($murder_{V,N}$) = *weapon;* A_1($anger_N$) = *angry;* Adv_1(*anger*) = *angrily*;

- Support verbs: $Oper_1$(*support*) = [*to*] *lend* [~ *to N*]; $Oper_1$(*promise$_N$*) = [*to*] *make* [ART ~]; $Func_2$(*proposal*) = *concerns* [*N*];
- Realization verbs: $Real_1$(*bus*) = [*to*] *drive* [ART ~]; $Real_2$(*bus*) = [*to*] *ride* [*on* ART ~]; $Real_1$(*promise$_N$*) = [*to*] *keep* [ART ~];
- Modifiers: Magn(*injury*) = *serious*; Ver(*citizen*) = *loyal*; Bon(*analysis*) = *fruitful*.

Complex LFs are formed by the combination of simple standard ones: for example, the LFs *Anti* and *Bon* can be combined to form the LF *AntiBon*: AntiBon(*hotel*) = {*seedy*, *sleazy*, *//flea bag*}. The symbol "//" before "flea bag" represents a fused element: the keyword *hotel* does not compose with the value of the LF function to form a collocation. Compare to: *seedy hotel*, *sleazy hotel*.

The advantage of using LFs for modeling relations between lexical units are many. We present here some of them, as stated by Kolesnikova (2011):

- LFs are universal. They represent semantic relations that are present in virtually all languages. This allows us to use them for building representations in several languages for multilingual alignments, to be used in automatic translation applications, multilingual information search, ontology alignment in different languages, etc;
- LFs are idiomatic. This allows the representation of a "non-typical" sense that emerges only when certain words are found together. For example, in English we can say "to know firmly". In this expression, the sense of "know firmly" is idiomatic. One can use the LF *Magn* (magnification or intensification) to represent this relation: *Magn* (*know*) = {*firmly*};
- Some LFs are the converse of one another, which can account for the paraphrase and passivization of collocations: *$Oper_1$* (*analysis*) = {[*to*] *carry out* DET ~} (*John carries out the analysis*); *$Func_1$* (*analysis*) = {DET ~ *is due* [*to*] N} (*The analysis is due to John*);

2.3 French Lexical Network

The *French Lexical Network* (FLN) (Lux-Pogodalla and Polguère, 2011) is based on the MTT, more specifically on the LFs. We extract from FLN the LFs that appear in lexical relations, in a total of about 100 simple LFs and 500 complex ones. The total number of LFs is elevated because, for instance, *$Oper_1$*, *$Oper_2$*, *$Oper_3$* and *$Oper_4$* are considered distinct LFs and there are many different complex LFs, for instance, *$CausFinOper_1$* and *$S_0SingReal_1$*.

FLN has been built manually by a lexicographic team of around 15 persons. Luxpogodalla and Polguère (2011) explain that lexicographic strategies used to extract linguistic information from corpora are based on the Explanatory Combinatorial Lexicology (Mel'čuk et al, 1995) and that they also make extensive use of the *Trésor de la Langue Française informatisé* (Dendien and Pierrel, 2003) as a lexical database from which to extract lexicographic information.

An important idea we extracted from the FLN is the concept of LF family. For example, *$Oper_1$*, *$Caus_1Oper_1$*, *$Caus_2Oper_1$*, *$Caus_3Oper_1$*, *$FinOper_1$*, etc., all belong to the LF family *$Oper_1$*.

2.4 Semantic Perspective

Jousse (2010) presents a system for the classification of lexical functions in four different ways: a semantic, a syntactic, a combinatorial and a paradigmatic classification. We present here the semantic classification, the only one we have included in our model, to this date.

In the semantic classification, LFs are divided in twelve classes: action-event, causativity, element-set, equivalence, form (or way), location, opposition, participants, phase-aspect, qualification, semantically-empty-verb and support-verb. Each class is divided in one or more sub-classes. For example, action-event has nine sub-classes: attempt, creation, decrease/degradation, imminence, manifestation, etc., and location has two: spatial/temporal and typical place.

Each LF has at least one meaning associated to it, and then each LF is classified in one or more semantic perspective. For example, *Magn* (intensification) is associated with the class "qualification", sub-class "intensity", while the LF *Bon* is also associated with the class "qualification", however with the sub-class "judgement".

2.5 The lemon Model

lemon (McCrae et al., 2012) is a model for sharing lexical information on the semantic web. It is based on earlier models, such as LexInfo (Buitelaar, 2009) and LMF (Francopoulo, 2007). As its main advantages over these previous models, we cite:

- separation between the linguistic and the ontological information;
- linguistic information, such as "partOfSpeech" and "writenForm" are represented as RDF properties, differently of LMF, which represent them as attributes of a property, which makes easier the use of other resources, like the SPARQL query language;
- lemon uses ISOCat, data categories homologated by ISO (for example, "partOfSpeech", "gender" and "tense");
- lemon is an easily extensible model;
- there are already many linguistic resources in lemon format, like WordNet and DBPedia Wiktionary.

Lexical units are represented in the lemon model using the classes "LexicalEntry" and "LexicalForm". The "LexicalEntry" class is connected to the lexical unit sense, which is represented by the "LexicalSense" class. The connection between the lemon model and external ontologies are made through this last class.

In our model, the keyword and the value of a LF will be represented as a lemon "LexicalSense" class. In MTT, the different senses of a word are represented by subscripts, using Roman and Arabic numbers and Latin letters (Mel'čuk, 1995), which we illustrate here with an example. Consider the word "*ocean*". It has concrete senses, like "*a body of water that covers the planet*" and abstract senses, like in "*ocean of people*". In MTT, the concrete senses of "*ocean*" would be represented as "Ocean_{I}" and the abstract senses as "Ocean_{II}". Inside "Ocean_{I}" we could have subdivisions:

- $\text{Ocean}_{\text{I.1a}}$: "*extension of water that covers the planet*" (always in singular);
- $\text{Ocean}_{\text{I.1b}}$: the set of oceans in general (always in plural) – "*the oceans are polluted*.";
- $\text{Ocean}_{\text{I.2}}$: a part of $\text{Ocean}_{\text{I.1a}}$ in a specific region – *Atlantic Ocean*, *Pacific Ocean*, *Arctic Ocean*, etc.

In our model, the word "*ocean*" is represented by a lemon object "LexicalEntry" and $Ocean_{\text{I}}$, $Ocean_{\text{I.1a}}$, $Ocean_{\text{I.1b}}$, $Ocean_{\text{I.2}}$ and $Ocean_{\text{II}}$ are each represented by a "LexicalSense" lemon object. The reason for this is explained as follows: the semantic connection represented by an individual LF is between senses, and not between lexical forms or lexical entries. By doing so, we can have an already disambiguated lexical network when connecting lexical units with a LF.

3 The lexfom Model

This section presents our model for the representation of LFs. The lexical function ontology model (lexfom) is divided in four modules: lexical function representation (lfrep), lexical function family (lffam), lexical function semantic perspective (lfsem) and lexical function relations (lfrel). Each subsection presents one of these modules.

3.1 The lfrep Module

Figure 1 illustrates the lfrep module and its connection to lffam and lfsem. The central class in this module is "lexical_function". In this figure and in the following ones, a black arrow represents an object property relation between classes and a white arrow represents a sub-class relation.

In yellow, we have classes which represent characteristics of a LF, e.g. if it is simple or complex, standard or semi-standard, etc. In grey, we have classes representing constituent of a LF. Most of those classes are specific to some LF families. For example, the "spatial specification" appear in the LF *Loc*: Loc_{ab}, Loc_{ad} and Loc_{in} and "intensification dimension" in the LF *Magn*: *Magn_behaviour*, *Magn_height*, *Magn_size*, etc.

In this module we can indicate to which LF family a LF belongs, using the object property "belongsToLFF" and we can also connect a LF to the meaning it denotes, using the object property "hasSemanticPerspective". Each of this properties connects to classes which belongs to different modules, *lffam* (lexical function family) and *lfsem* (lexical function semantic perspective), respectively, which will be presented in the next two sections.

Figure 2(a) illustrates the part of the lfrep module used to represent complex LFs and Figure 2(b) shows an example of a complex LF (*AntiBon*) represented in RDF/OWL format (turtle dialect), following the general schema of Figure 2(a).

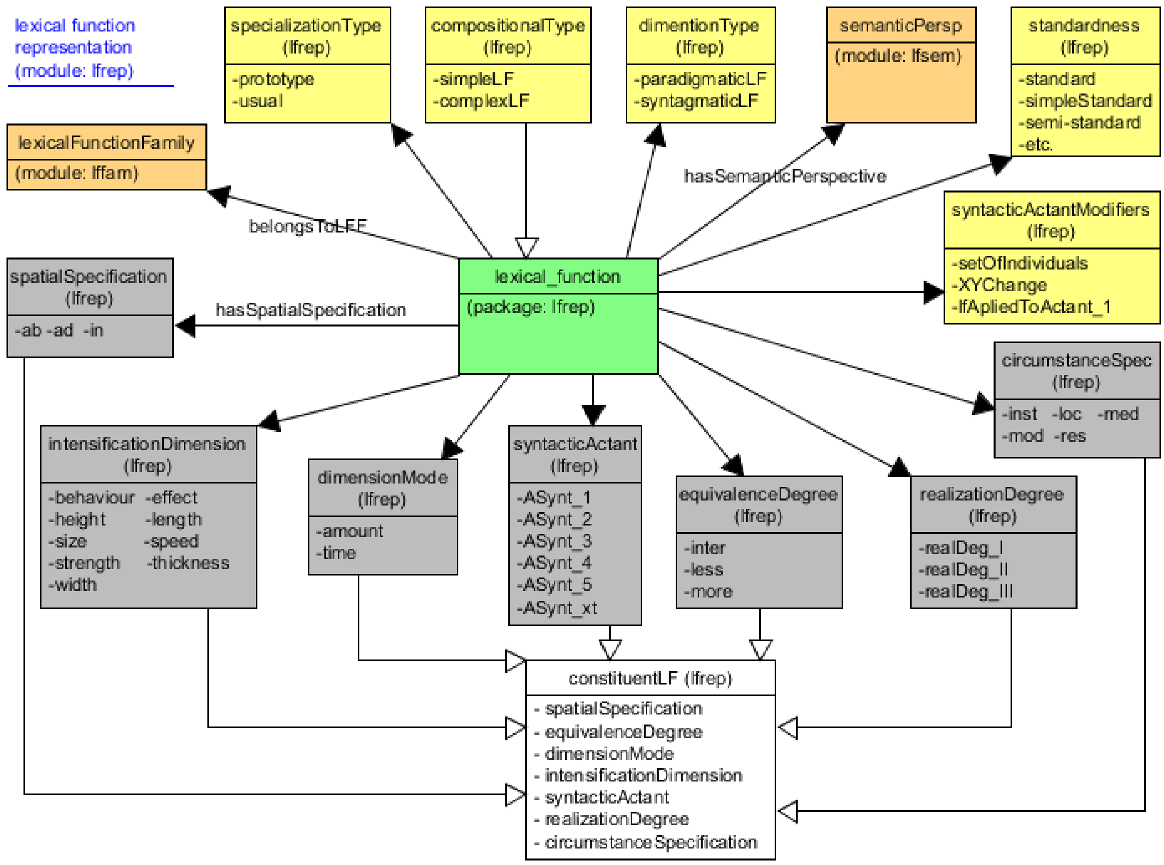


Figure 1: Module to represent a lexical function (lfrep) and its connection to lfsem and lffam.

3.2 The lffam Module

Figure 3 illustrates the module *lffam*. This module is used to cluster the LFs into families, according to their semantic/syntactic similarity, as introduced in Section 2.3. There are two main groups, the paradigmatic and the syntagmatic LF. The last one is subdivided in support verbs, phasal verbs (which indicate the start, continuation or finalization of an action), realization verbs, causation verbs (cause, permission and liquidation of an action) and some other functions which are not classified in any specific sub-group. The paradigmatic LFs are subdivided in nine groups, e.g. actantial nouns, adjectives and adverbs, circumstantial nouns, syntactic conversion, etc.

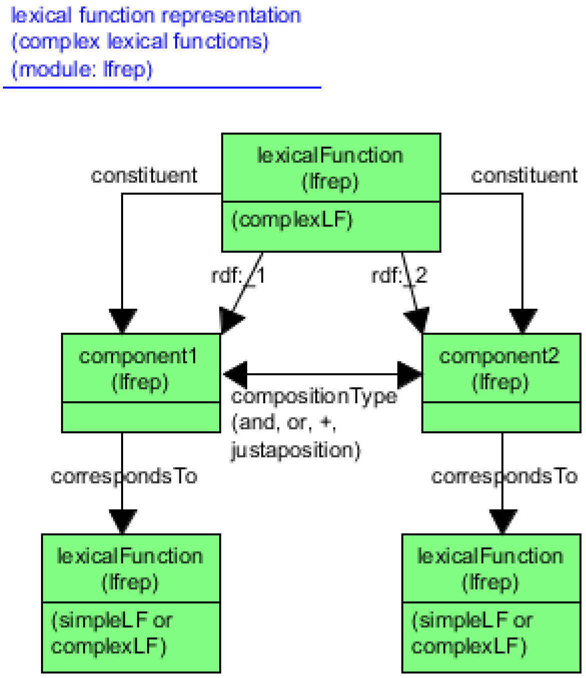
The concept of family was extracted from FLN. Complex LFs whose base LF is the same belong to the same family, as explained in Section 2.3. FLN contains about 100 LF families, each one roughly corresponding to a simple LF. It is important to note that similar LFs, differentiable one from another only by their syntactic actant, such as $Oper_1$, $Oper_2$, $Oper_3$, etc, is each one the head of a LF family. For example, we have the $Oper_1$ family ($Oper_1$, $Caus_1Oper_1$, $Caus_2Oper_1$), the $Oper_2$ family ($Oper_2$, $Caus_1Oper_2$, $Caus_2Oper_2$, $Caus_3Oper_2$, $FinOper_2$, etc.) and so on.

3.3 The lfsem Module

Figure 4 illustrates the LF semantic perspective module, as introduced in Section 2.4. Twelve semantic classes are represented, each one divided in one or many sub-classes. A LF can have one or more semantic perspective, depending on the context and on the lexical units it connects.

For example, the LF *Magn* (intensification) has the semantic perspective "qualification/intensity" and the LF *Syn* (synonymy) has the semantic perspective "equivalence/similar lexical units".

Figure 5 shows a RDF representation of the LF *Anti*. Note how the modules "lfrep", "lffam" and "lfsem" are used to represent it.



(a) Part of the lfrep module used to represent complex LFs.

```
lfrlf:LF-AntiBon rdf:type
    lfrep:complexLF;
lfrep:constituentComplexLF
    :CCLF-AntiBon-Anti,
          CCLF-AntiBon-Bon.

:CCLF-AntiBon-Anti rdf:type
    lfrep:componentComplexLF;
lfrep:justapositionComposition
    :CCLF-AntiBon-Bon;
lfrep:correspondsTo lfrlf:LF-Anti.

:CCLF-AntiBon-Bon rdf:type
    lfrep:componentComplexLF;
lfrep:correspondsTo lfrlf:LF-Bon.
```

(b) RDF/OWL representation of the complex LF *AntiBon*, using the module lfrep.

Figure 2: Representation of complex LFs.

3.4 The lfrel Module

Figure 6 illustrates the lexical function relation module (lfrel), which represents the way lexical units are connected by a LF.

We decided to connect the LF keyword and the LF value using an intermediate class (lfSenseRelation), which is a subclass of a the lemon class "SenseRelation", instead of connecting them directly with the LexicalFunction class because in this way we can connect to the lfSenseRelation information that is specific to the relation between two lexical units, independently of the LF connection them, and we can connect to a LF information that is independent of the lexical units that it connects. Also, the paradigmatic/syntagmatic information (LRType) is connected to the LexicalRelation class instead of being connected to the LexicalFunction class.

Although the LFs usually have a definite type (paradigmatic or syntagmatic), some of them do not have it, which will depend on the lexical units they model.

The property "hasGovPattern" is used to indicate the government pattern (GP) in the sense/lexical relation. For example, the collocation "receive an order from N", is modeled by the LF $Oper_3$(*order*$_N$) = *receive* and its GP is [ART ~ *from* N]. For the moment, the GPs are represented by strings in our model, but we intend to create a module "lfgpat" with a hierarchy of the most commons GPs.

Figure 7 illustrates how the collocation "*close friend*" can be represented. It is modelled by the LF Magn (predicative sense = intensification): Magn(*friend*$_{I.1}$) = *close*$_{III.1a}$; Since also Magn(*friend*$_{I.1}$) = *good*$_{II}$, we could have another LexicalRelation (*Magn_02*) connecting the LexicalSense *good*$_{II}$ and the LexicalSense *friend*$_{I.1}$.

The lexical relation is connected to the value of the collocation using the property "hasLFValue" and to the keyword using the property "hasLFKeyword". The property "hasLRType" informs that the relation between "*close*$_{III.1a}$" and "*friend*$_{I.1}$", modelled by the LF "Magn", is a syntagmatic relation.

As explained in Section 2.5, it is important to note that the lexical units that appear in our example, "*friend*$_{I.1}$", and "*close*$_{III.1a}$" will be modeled as "LexicalSense" and not as a "LexicalEntry" lemon object. This means that our model will connect to the lemon model via the sense of the lexical units. This allows the construction of already disambiguated lexical networks. Finally, the lexical variations (e.g. plural) can be treated at the level of the LexicalEntry lemon object, already implemented by lemon.

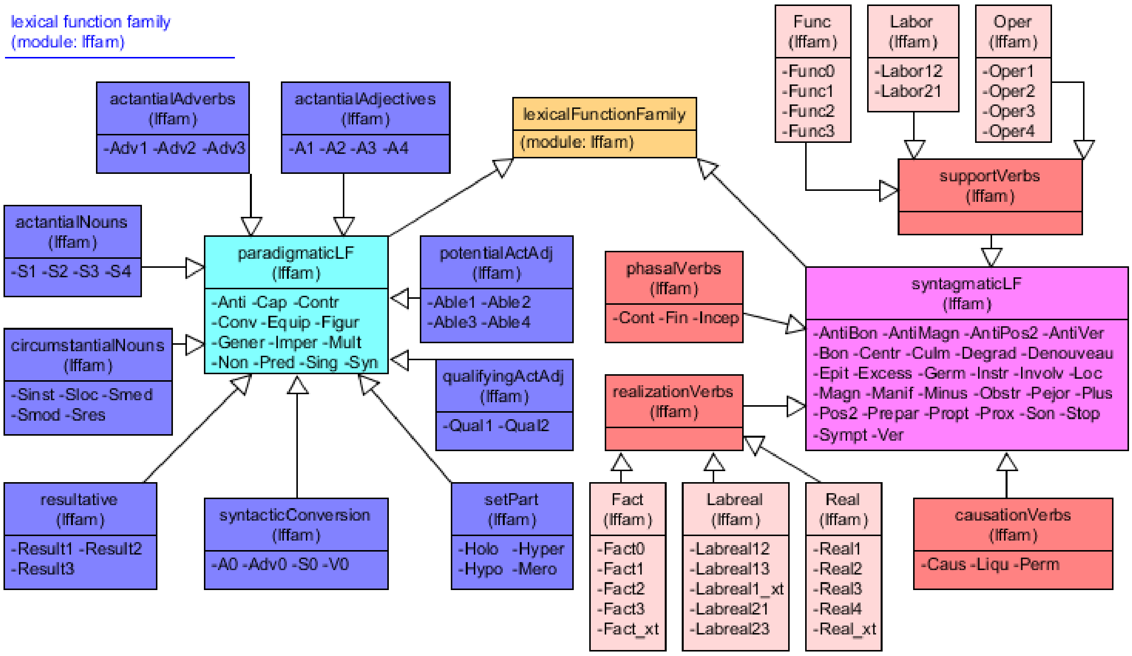


Figure 3: Module lffam, which represents the lexical functions families.

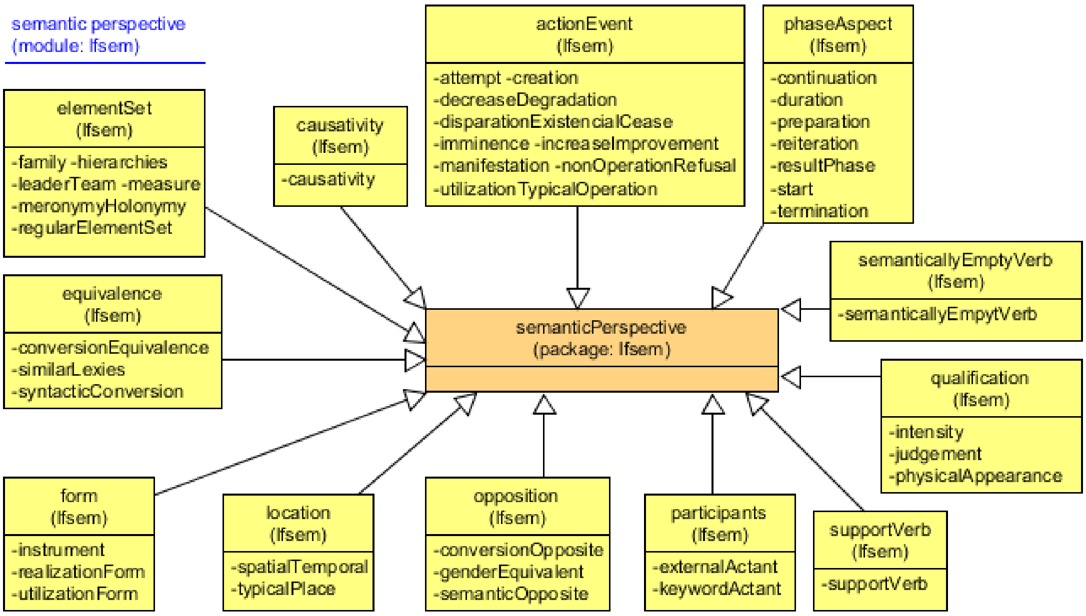


Figure 4: Semantic perspective module (lfsem) contains twelve classes used to indicate the semantics of a LF.

One advantage of representing lexical units as lemon LexicalSense/LexicalEntry is that lemon implements the Syntax and Semantics (synsem)[1] module, which can be used to connect its LexicalEntry class to syntactic and morphological information about lexical units.

For example, we can use the property "syntacticBehavior" to indicate that a lexical unit is a transitive verb, to indicate its direct object, etc. We can also indicate alternative spelling of a lexical unit (e.g. American/British spelling). lemon also implements the Variation and Translation (vartrans)[2], which can be used to connect a lexical unit to its translations in other languages.

Another advantage of using lemon is the following: it implements the connection of a LexicalSense to a concept defined by an external ontology, such DBPedia[3], through the "reference" property.

[1] https://www.w3.org/2016/05/ontolex/#syntax-and-semantics-synsem
[2] https://www.w3.org/2016/05/ontolex/#variation-translation-vartrans
[3] http://dbpedia.org/ontology/

```
lfrlf:LF-Anti rdf:type lfrep:simpleLF;
     lfrep:belongsToLFFamily lffam:LFF-par-Anti;
     lfrep:dimension lfrep:type-paradigmaticLF;
     lfrep:semanticPerspective lfsem:pSem-op-semanticOpposite.
```

Figure 5: RDF representation of the simple LF *Anti*, using the modules lfrep, lffam and lfsem.

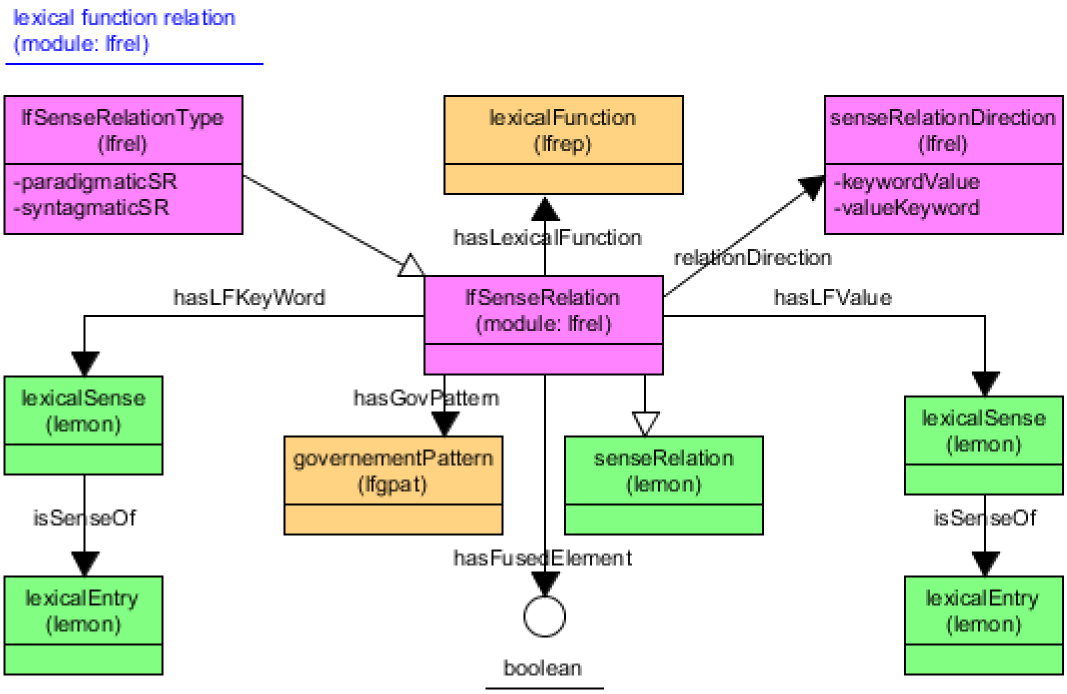


Figure 6: Lexical function relation module (lfrel).

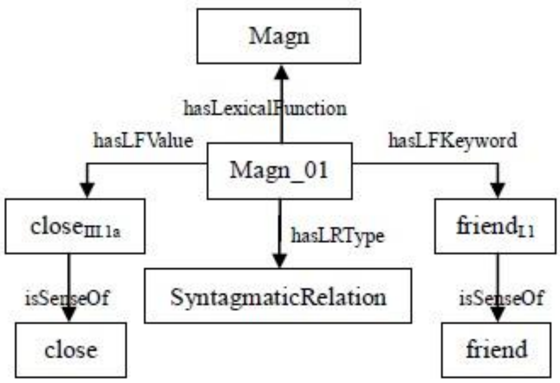


Figure 7: Representation of the collocation “close friend” using the lfrel module.

4 Application

The information about LFs, lexical units and the syntagmatic relation between lexical units were extracted from FLN. In this lexical network, this information is stored in a relational data bank and can be extracted by SLQ table dumps or SQL queries.

FLN has about 600 standard LFs (about 100 simple and 500 complex LFs) and we encoded all of them using lexfom.

We also encoded the lexical relations mediated by these standard LFs. So far we have encoded using the lexfom model about 54,000 relations, being 46,000 paradigmatic and 8,000 syntagmatic.

Figure 8 shows the RDF representation of the French collocation *porter un vêtement* (*to wear a clothe*). It is intermediated by the LF $Real_1$: Real_1 (*vêtement*) = {*porter*}. Each word is first represented as a lemon:LexicalEntry, and each sense of the words are represented as a lemon:LexicalSense. The

LF *Real*$_1$ is represented with its properties, using the modules *lfrep*, *lffam* and *lfsem*. Finally, the module *lfrel* is used to represent the sense relation between *porter* and *vetêment*.

```
:lex_vêtement a ontolex:LexicalEntry,
                ontolex:Word;
 ontolex:canonicalForm:form_vêtement;
 ontolex:sense :vêtement_sense_I.2;
 ontolex:sense :vêtement_sense_I.1;
 ontolex:sense :vêtement_sense_II;
 ontolex:sense :vêtement_sense_III.1;
 ontolex:sense :vêtement_sense_III.2;
 rdfs:label "vêtement"@fr.
```

```
:form_vêtement a ontolex:Form;
    ontolex:writtenRep "vêtement"@fr .

vêtement_sense_I.2   a ontolex:LexicalSense.
vêtement_sense_I.1   a ontolex:LexicalSense.
vêtement_sense_II    a ontolex:LexicalSense.
vêtement_sense_III.1 a ontolex:LexicalSense.
vêtement_sense_III.2 a ontolex:LexicalSense.
```

```
:lex_porter a ontolex:LexicalEntry,
  ontolex:Word;
  ontolex:canonicalForm :form_porter;
  ontolex:sense :porter_sense_I;
  ontolex:sense :porter_sense_II;
  rdfs:label "porter"@fr.
```

```
:form_porter a ontolex:Form;
  ontolex:writtenRep "porter"@fr .

porter_sense_I a ontolex:LexicalSense.
porter_sense_II a ontolex:LexicalSense.
```

```
LF-Real1 rdf:type lfrep:simpleLF,
     lfrlf:Real1,owl:NamedIndividual;
lfrep:belongsToLFFamily
  lffam:LFF-synt-realV-Real1;
lfrep:hasSyntActant
  lfrep:lfrep-const-sa-ASynt_1;
lfrep:dimension
  lfrep:lfrep-type-syntagmaticLF;
lfrep:semanticPerspective
  lfsem:pSem-ae-utilTypicalOperation.
```

```
:lfsr_11420 a
           lfrel:SyntagmaticLFSenseRelation;
   lfrel:hasLexicalFunction lfrep:LF-Real1;
 lfrel:hasLFKeyword
           ontolex:vêtement_sense_I.2;
 lfrel:hasLFValue ontolex:porter_sense_IV;
 lfrel:hasGovPattern
           lfgpat:"DET ~s"^^xsd:string;
lfrel:relationDirection lfrel:valueKeyword;
lfrel:hasFusedElement "false"^^xsd:boolean.
```

Figure 8: RDF representation of the LF *Real*$_1$, the lexical units *vêtement* and *porter* and the sense relation between *vêtement* and *porter* to form the French collocation *porter un vêtement*.

5 Conclusion and Future Work

We present in this paper an ongoing project, called Lexical Functions Ontology Model (lexfom), aimed at the representation of the lexical functions of Meaning-Text Theory as a lexical ontology.

Most of the existing lexical networks lack important semantic information, especially the syntagmatic relations between lexical units. Lexical functions are a powerful tool for the representation of linguistic relations. In particular, syntagmatic lexical functions can fill the present gap in the representation of syntagmatic relations in lexical networks.

Moreover, the combination of the descriptive logic embedded in the OWL language with the semantic, syntactic, paradigmatic and combinatorial information, provided by lexical functions, creates a strong tool for studying human reasoning, the relation between lexical units and can be used by diverse natural language processing applications and tools.

Finally, this work can be seen as a new form of representation of collocations. It is important to observe that we deal with collocations as defined by Mel'čuk (1998), and not the definition usually employed in NLP articles, which usually states a collocation as "word cooccurrences whose idiosyncrasy is of statistical nature only" (Vincze et al., 2016).

Dealing with collocations, as stated in Section 2.1, is of vital importance for a real understanding and correct identification and representation of the relations between lexical units.

As a future work, we intend to use our model to transform the *French Lexical Network*, from its present relational database format to an ontology format. We have so far encoded about 100 simple LFs and 500 complex LFs, extracted from FLN, and also encoded about 54,000 lexical relations, being 46,000 paradigmatic and 8,000 syntagmatic relations.

Also, similarly to the lfsem module, new modules will be created to represent the remaining classifications presented by Jousse (2010): a syntactic, a combinatorial and a paradigmatic classification module.

Finally, we intend to combine the semantic information in our ontology with a word embeddings model to enhance the automatic construction of lexical networks.

References

Bolshakov, I. and Gelbukh, A. (1998). Lexical functions in Spanish. *Proceedings CIC-98, Simposium Internacional de Computación*, pp. 383-395. November 11-13, 1998, Mexico D.F., Mexico.

Buitelaar, P., Cimiano, P., Haase, P. et Sintek, M. (2009). Towards linguistically grounded ontologies. In *L. Aroyo et al. (Eds.): ESWC 2009, LNCS 5554*, pp. 111-125, Spring-Verlag Berlin. Heidelberg 2009.

Dendien, J. and Pierrel, J.-M. (2003). Le Trésor de la Langue Française informatisé: un exemple d'informatisation d'un dictionnaire de langue de référence. Traitement Automatique des Langues (T.a.l.), 44(2):11–37.

Francopoulo, G., Bel, N., Georg, M., Calzolari, N., Monachini, M., Pet, M. and Soria, C. (2007). Lexical markup framework: ISO standard for semantic information in NLP lexicons. In: *Proceedings of the Workshop of the GLDV Working Group on Lexicography at the Biennial Spring Conference of the GLDV*.

Jousse, A. (2010). Modèle de structuration des relations lexicales fondé sur le formalisme des fonctions lexicales. Thèse de doctorat. Université de Montréal et Université Paris Diderot (Paris 7), 340 p.

Kolesnikova, O. Automatic extraction of lexical functions, PhD Thesis (2011). Instituto Politecnico Nacional – Centro de Investigacion en Computacion, Mexico, D.F., Mexico, 116 p.

Lux-Pogodalla, V. and Polguère, A. (2011). Construction of a French lexical network: Methodological issues. In *Proceedings of the First International Workshop on Lexical Resources, WoLeR'11.* An ESSLLI 2011 Workshop, pp. 54-61, Ljubljana, Slovenia.

Manning, C. D. and Schütze, H. (1999). Foundations of Statistical Natural Language Processing. Cambridge, MA: The MIT Press, 1999, 680 p.

McCrae, J., Aguado-de-Cea, G., Buitelaar, P., Cimiano, P., Declerck, T., Gómez-Pérez, A., Gracia, J., Hollink, L., Montiel-Ponsoda, E., Spohr, D. and Wunner, T. (2012). Interchanging lexical resources on the Semantic Web. *Lang Resources & Evaluation (2012)* 46:701–719.

Mel'čuk, I. (2004). Actants in semantics and syntax, I: Actants in semantics. *Linguistics* 42(1):1-66.

Mel'čuk, I. (1998). Collocations and lexical functions - *A.P. Cowie (Ed.), Phraseology. Theory, Analysis and Applications*, Oxford: Clarendon Press, pp. 23-53.

Mel'čuk, I. (1997). Vers une linguistique Sens-Texte. Leçon inaugurale. Paris: Collège de France, 78 p.

Mel'čuk, I. (1996). Lexical functions: A tool for the description of lexical relations in the lexicon. In *L. Wanner (ed.): Lexical Functions in Lexicography and Natural Language Processing*, pp. 37-102, Amsterdam/Philadelphia: Benjamins.

Mel'čuk, I., Clas, A. et Polguère, A. (1995). *Introduction à la lexicologie explicative et combinatoire*, Coll. Champs linguistiques/Universités francophones, Louvain-la-Neuve/Paris, Éditions Duculot/AUPELF-UREF, 256 p.

Mel'čuk, I. (1992). Paraphrase et lexique: La théorie Sens-Texte et le *Dictionnaire explicatif et combinatoire* in Mel'cuk *et al*. 1992: 9-58.

Morgan, J. L. (1978), 'Two types of convention in Indirect Speech acts', in P. Cole (ed.), *Syntax and Semantics, v.9. Pragmatics* (New York *etc*.: Academic Press), 261-280.

Polguère (2000). Towards a theoretically-motivated general public dictionary of semantic derivations and collocations for French. In *Approaches to Lexical Combinatorics, Proceedings of Euralex 2000*, p. 517-527.

Heid, U. et Raab, S. (1989). Collocations in Multilingual Generation. In *Proceedings of the fourth conference on European chapter of the Association for Computational Linguistics* - EACL '89, pp. 130-136. Stroudsburg, PA, USA.

Saussure, F. de (1983). Course in general linguistics. Eds. Charles Bally and Albert Sechehaye. Trans. Roy Harris. La Salle, Illinois: Open Court. 1983, 236p.

Schwab, D., Tze, L. L. et Lafourcade, M. (2007). Les vecteurs conceptuels, un outil complémentaire aux réseaux lexicaux. TALN'07: Traitement Automatique des Langues Naturelles, pp. 293-302, Jun 2007, Toulouse, France, ATALA.

Vincze, V., Savary, A., Candito, M. and Ramisch, C. (2016). Annotation guidelines for the PARSEME shared task on automatic detection of verbal MultiWord Expresions. Version 5.0. http://typo.uni-konstanz.de/parseme/images/shared-task/guidelines/PARSEME-ST-annotation-guidelines-v6.pdf