ISSN 2007-9737

Automatic Detection of Lexical Functions in Context

Olga Kolesnikova

Instituto Politécnico Nacional,
Escuela Superior de Cómputo,
Mexico

kolesolga@gmail.com

Abstract. In this article, we explore the concept of lexical functions as a formalism to represent recurrent lexical, semantic, and syntactic relations among words. A lexical function takes a word as input and outputs a set of words related to the input in a certain way on the paradigmatic or syntagmatic level. For example, the syntagmatic lexical function termed Oper1 takes the noun *decision* as input and outputs the verb *make* with the semantics of 'Agent realizes the action denoted by the noun', so Oper1 captures the relation between the noun and the verb in the collocation *make a decision*. The numeric part of the Oper1 notation reflects two facts: first, the action of *make* is performed by the agent, which is viewed as the first semantic actant (or argument) in the Tesnière's model, and second, the syntactic function of the word denoting the agent in utterances with *make a decision* is subject. In general, lexical functions represent common semantic and syntactic patterns typical for certain word classes and can aid in many tasks of natural language processing, lexical and syntactic disambiguation being the most fundamental one among them. In this article, we review various paradigmatic and syntagmatic lexical functions, their application and identification in natural language processing.

Keywords. Collocation, lexical function, paradigmatic relations, syntagmatic relations, machine learning.

1 Introduction

Computational Linguistics or Natural Language Processing (NLP) is a significant and rapidly developing research area on the crossroads of linguistics and computer science. It deals with any issue, which involves human-computer interaction in a human (natural) language: speech recognition, information retrieval, question answering, opinion, sentiment and polarity identification, knowledge extraction, machine translation, among others.

The NLP fundamental challenges are ambiguity of language items and lexicon diversity. To combat these problems, various formal semantic concepts and models have been developed. In this article, we discuss one of these: the concept of lexical function. This formal tool was created, has been developed and implemented in order to assist in solving both challenges mentioned above.

To explain it informally, lexical function captures lexical, semantic, and syntactic features common to sufficiently big groups of words and phrases. The different types of lexical functions form a taxonomy allowing representing and classifying in a consistent way the variety of words and word combinations in a natural language.

In this article, we speak of what a lexical function is, briefly consider several types of lexical functions, illustrate and exemplify the lexical function typology using verbal lexical functions, which are probably most richly represented in natural languages, and discuss application and detection of lexical functions in language systems.

2 Lexical Function

Lexical function is a formalism proposed by Mel'čuk and Žolkovskij [23] and exposed in detail in [21] to represent various relations among words. Originally, the concept of lexical function was created with the purpose to generalize typical lexical relations; however, it is often interpreted as a categorization of lexical items according to their lexical, semantic, and syntactic features [8].

A lexical function takes a word as input and outputs a set of words related to the input in a certain way on the paradigmatic or syntagmatic

doi: 10.13053/CyS-24-3-3774

level. It is a function in the mathematical sense defined as a mapping from a word w_0 called the lexical function argument to the lexical function value, which is a set of words $\{w_1, w_2, \ldots w_n\}$ where each word w_i, $1 \leq i \leq n$, has a particular (and the same) lexical relation with the argument w_0; so using mathematical notation, lexical function (LF) is represented as LF(w_0) = $\{w_1, w_2, \ldots w_n\}$.

Wanner [34] states that lexical function is a concept, which can be used to systematically describe "institutionalized" lexical relations clarifying that "a lexical relation is institutionalized if it holds between two lexical units L_1 and L_2 and has the following characteristics: if L_1 is chosen to express a particular meaning M, its choice is predetermined by the relation of M to L_2 to such an extent that in case M and L_2 is given, the choice of L_1 is a language-specific automatism."

Institutionalized lexical relations can be of two types: paradigmatic and syntagmatic. Paradigmatic relations are observed between lexical units within a lexicon (examples of paradigmatic relations are synonymy, antonymy, hypernymy, hyponymy, etc.) and syntagmatic relations hold between lexical units that co-occur in texts (*make a decision*, *friendly attitude*, *rain cats and dogs*).

Let us give some examples of institutionalized lexical relations:

- *chair* – *furniture* (*is-a* or hypernymy-hyponymy relation),
- *laugh* – *laughter* (relation of derivation of a noun from a verb),
- *spot* – *rain* (relation between a substance and its unit),
- *discussion* – *general* (relation between an act and its feature with respect to coverage),
- *argument* – *reasonable* (relation between a concept and its feature with respect to quality),
- *build* – *house* (relation between an action involving causation and creation applied to a physical object),
- *lesson* – *give* (relation between an action of realization applied to a communication object).

In examples (1-3), the institutionalized lexical relations are paradigmatic and in examples (4-7), they are syntagmatic. The *is-a* relation in example (1), also known as hypernymy-hyponymy relation, is represented by the lexical function denominated Gener (from Lat. *genus*, category). Gener takes a word with a more specific meaning as its argument and outputs the word with a more generic meaning, so using the functional notation we get Gener(*chair*) = *furniture*.

The words in example (2) are related by means of nominalization of the verb, and this phenomenon is captured by the lexical function S0 (S is from Substantival): S0(*laugh*) = *laughter*. In the denomination of this function, two elements are present: the letter S which reflects the semantic aspect of the relation between *laugh* and *laughter*, and the number zero.

Numbers are employed in lexical function nomenclature to represent positions within the valency structure of the word, which is the lexical function value. These positions are called actants in the language model developed by Lucien Tesnière, the founder of the dependency grammar, see an explanation of dependency grammar and dependency parsing in [25]. Actants are sometimes called participants of a situation in semantics and cognitive sciences, e.g. [10], and the typical term for an actant in natural language processing is semantic or thematic role [30].

In the example of S0 given above, zero represents the fact that *laughter* is the name of the situation, therefore, information of its actants is irrelevant in the relation between the action verb *laugh* and the corresponding action noun *laughter*.

Another lexical function called IncepOper1 (from Latin *incipere*, begin, and *operari*, do) outputs a set of one or more verbs with the meaning 'do, carry out, perform, experience', so if its argument is *laughter*, IncepOper1 generates as its value the set {*burst into*, *explode into*, *break into*}, i.e. IncepOper1(*laughter*) = {*burst into*, *explode into*, *break into*}; 1 reflects the fact that the first actant (agent) of the situation *laughter* has the syntactic function of subject in sentences and clauses where *laugher* is used in combination with its IncepOper1 verb, e.g. *It looked so funny that we burst into laughter*.

Therefore, IncepOper1 represents the semantic and syntactic pattern 'Agent performs the action denoted by the noun'. In the next section, we discuss various types of lexical functions.

ISSN 2007-9737

Table 1. Paradigmatic lexical functions

Lexical function	Examples
Syn corresponds to the basic semantic relation of synonymy. According to various types of the synonymy relations, the following lexical functions are recognized: Syn, Syn$_{\supset}$, Syn$_{\subset}$, Syn$_{\cap}$.	
Syn represents equivalency relation	Syn(*myopia*) = *nearsightedness* Syn(*wealthy*) = *rich* Syn(*telephone*) = *phone* Syn(*airplane*) = *plane*
Syn$_{\supset}$ represents synonyms with more specific, richer, narrower meaning	Syn$_{\supset}$(*realize*) = *notice* Syn$_{\supset}$(*fire upon*) = *shell*
Syn$_{\subset}$ represents synonyms with less specific, poorer, broader meaning	Syn$_{\subset}$(*feeling*) = *emotion* Syn$_{\subset}$(*laugh*) = *giggle*
Syn$_{\cap}$ represents synonyms with intersecting meaning	Syn$_{\cap}$(*say*) = *tell* Syn$_{\cap}$(*escape*) = *elude*
Conv*klj* (converse = reversed in order or relation). The value of this function is a word denoting the same event as the argument but with its actants permutated. For example, Conv$_{21}$ means that the first (agent) and the second (patient) actants are permutated, i.e. the second actant becomes the first actant, and the first actant becomes the second one.	Conv21(*fear*) = *frighten* Conv21(*husband*) = *wife* Conv21(*buy*) = *sell* Conv21(*behind*) = *in front of* Conv21(*include*) = *belong* Conv321$_1$(*teach*) = *learn* Conv$_{321}$ means that the third actant become the first one, the first actant becomes the second one, and the second actant becomes the third one. The actants of teach are (1) teacher, (2) learner, (3) subject, theme as in *He taught me algebra*.
Anti. The value of this function is the antonym of its argument.	*male* – *female*, *long* – *short*, *up* – *down*, *precede* – *follow*, *joy* – *grief*, *accept* – *reject*, *start* – *stop*
Der (syntactic derivative). The value is a word with the same meaning as the argument, but belongs to another grammatical category or part of speech. Four basic parts of speech are distinguished: S – noun, A – adjective, Adv – adverb, and V – verb, so there are four derivatives: S0, A0, Adv0 and V0.	S0(*employ*) = *employment* S0(*apply*) = *application* A0(*music*) = *musical* A0(*water*) = *watery* Adv0(*friend*) = *friendly* Adv0(*good*) = *well* V0(*code*) = *codify* V0(*commerce*) = *commercialize*
Gener (from Lat. *genus*, class, category). The value of the closest generic concept of the argument, e.g. Gener(*gas*) = *substance*.	*strawberry* – *berry*, *linguist* – *researcher*, *yellow* – *color*, *whisper* – *speak*, *crawl* – *move*, *carrots* – *vegetables*
S*i* (*i* = 1, 2, 3, 4) is the word which is the standard name of the *i*th actant of the argument.	S1(*examine*) = *examiner* S1(*play*) = *player* S2(*examine*) = *examinee* S2(*letter*) = *addressee* S3(*lecture*) = *theme* S3(*letter*) = *contents* S4(*sell*) = *price*
S*c* is a typical name of a circumstantial component of the event (argument to this function): location (*loc*), instrument (*instr*), manner (*mod*), result (*res*), means (*med*).	S*loc*(*play*) = *theatre* S*instr* (*struggle*) = *weapon* S*mod*(*life*) = *way* (of life S*res*(*split*) = *crack* S*med*(*teach*) = *teaching materials*
Mult (from Latin *multum*, multitude) is a typical name of a collection of entities denoted by the argument, Mult captures the meronymy (part-whole) relation.	Mult(*cattle*) = *herd* Mult(*bee*) = *swarm* Mult(*dog*) = *pack* Mult(*vehicle*) = *fleet*
Sing (from Latin *singulus*, separate) is a typical name of a unit of an entity (argument of this function), Sing represents the meronymy relation like Mult, but viewed in the opposite direction (whole-part).	Sing(*rain*) = *drop* Sing(*snow*) = *snowflake* Sing(*violence*) = act (of *violence*)

doi: 10.13053/CyS-24-3-3774

tabl. Syntagmatic lexical functions whose value is a verb and the argument is a noun which is the name of an event; *sb* stands for somebody, *sth* stands for something, *A* stands for argument of a lexical function; examples are word combinations, which include the argument and the value of a lexical function

Lexical function	Examples
Oper$_i$ is from Lat. *operari*, do, carry out, perform, experience. The argument of Oper$_i$ is the name of event (action, activity, state, property), i.e. a predicative noun. The value of Oper$_i$ is a light verb connecting the name of the *i*th participant of the event functioning as the subject in an utterance with the name of the event functioning as a direct object.	
Oper1. The value means ‘perform, do, act’.	*lend support*, *put up resistance*, *give an order*, *ask a question*, *make a proposal*, *pay attention*
Oper2. The value means ‘undergo, meet’.	*receive support*, *meet resistance*, *receive the order*, *take an exam*, *undergo an analysis*
Func$_i$ is *from* Lat. *functionare*, function. The argument of Func$_i$ is the name of event (action, activity, state, property), i.e. a predicative noun, and the value means ‘occur, take place’.	
Func0. The value is a light verb meaning ‘happen, take place’, and the name of the event functions as the grammatical subject.	*snow falls*, *silence reigns*, *the smell lingers*, *the arrow flies*, *thunder roars*, *the war is on*, *the event takes place*
Func1. The value is a light verb meaning ‘originate from’, it connects the name of the event as the grammatical subject with the name of the first participant as a direct object.	*the blow comes from sth*, *the proposal comes/stems from sth*, *support comes from sb*
Func2. The value is a light verb meaning ‘concern, apply to’ and connecting the name of the event as the grammatical subject with the name of the second participant as a direct object.	*the blow falls upon sth*, *the proposal concerns sth*, *the analysis concerns sth*, *the analysis covers sth*
Labor$_{ij}$**(*k*)** is *from* Lat. *laborari*, to work, toil, process. The value is a light verb connecting the *i*th participant of the event as the grammatical subject with the *j*th participant as the direct object and with the argument of this function (the name of the event) as an indirect object.	
Labor12	*keep sth under control*, *treat sb with respect*, *subject sb to punishment*, *hold sb in high esteem*, *cut sth with a saw*, *submit sth to an analysis*, *to take sth into consideration*
Incep is from Lat. *incipere*, begin. The value is a verb meaning ‘start doing sth’.	
IncepOper1	*open fire on sth*, *acquire popularity*, *sink into despair*, *contract a disease*, *acquire/develop/form a habit*, *run into trouble*, *turn attention to sth*
IncepOper2	*fall under the power of sb*, *get under sb’s control*
IncepFunc1. The value means ‘sth begins to be experienced’.	*despair creeps over/in sb*, *hatred stirs up sb*, *anger arises in sb*

3 Types of Lexical Functions

In this section, we consider three typologies of lexical functions. The first one categorizes lexical functions depending on the vertical or horizontal orientation of relation among words; with respect to this feature, paradigmatic and syntagmatic lexical functions are distinguished as explained in Section 2.

The second typology classifies lexical functions depending on the complexity of their internal semantic structure: according to this criterion, lexical functions can be simple and complex.

The third typology distinguishes lexical functions considering if they are applicable to broad or narrow word classes: here lexical functions can be standard and non-standard.

Semantic relations between words can be observed on the vertical and horizontal axes. In Section 2 we described the relations informally, i.e., relations between words within the lexicon are paradigmatic (synonymy, antonymy, hypernymy, hyponymy, etc.), and relations between words in utterances are syntagmatic (dependency, constituency, syntactic functions, semantic roles, etc.).

Formally, Murphy [24] defines paradigmatic relation as the relation between words, which form a paradigm. For example, a semantic paradigm is a set of words belonging to:

Table 3. Syntagmatic lexical functions (continuation of Table 2)

Lexical function	Examples
Cont is from Lat. *continuare*, continue. The value is a verb meaning ‘continue doing sth’.	
ContOper1. The value means ‘continue to experience sth’.	*maintain enthusiasm*, *hope burns*, *anger boiled over in sb*
ContOper2	*hold attention*
ContOper12	*feel animosity towards*/*against sth*
ContFunc0	*the offer stands*, *the odor lingers*
ContFact0	*her luck holds*
Fin is from Lat. *finire*, cease. The value means ‘terminate doing sth’.	
FinOper1	*lose one’s power over sth*, *lose patience*, *lose influence*, *drop*/*get rid of a habit*
FinOper2. The value means ‘cease to be the object of sth, the latter being an aspect/characteristic of sb’.	*lose credit with sb*
FinFunc0. The value means ‘the argument of FinFunc0 ceases to be experienced’.	*anger defuses*, *hatred ceases*, *enthusiasm wanes*
FinFunc1	*his love vanished into thin air*
Caus*i* is from Lat. *causare*, cause. The value means ‘the ith participant of an event causes sth so that the event begins occurring’. Caus has the causative meaning without reference to a participant of the event.	
CausOper1	*lead sb to the opinion*, *throw sb into despair*
CausOper2	*put sth under sb’s control*, *call sth to sb’s attention*
Caus1Oper2	*bring sth under one’s control*
CausFunc0. The value means ‘cause the existence of the event’.	*bring about the crisis*, *create a difficulty*, *hold elections*
Caus1Func2	*sow suspicions*, *show attention to sb*
Caus2Func1. The value means ‘cause sth to be experienced’.	*raise hope in sb*, *give surprise*, *provoke anger*
Caus2Func2	*grab sb’s attention*, *seek friendship*
CausReal1	*fall under suspicion*
Caus1Manif	*produce admiration*, *show joy*, *enjoy friendship*
CausDegrad	*joy was vanishing*, *the fire is dying down*, *my strength is failing*
Perm*i* is from Lat. *permittere*, permit. The value means ‘the *i*th participant of an event permits/allows the event to occur or does nothing which would cause that the event ceases to occur’. Perm has the same permissive meaning without reference to a participant of the event.	
PermFunc1	
Perm1Fact0	*let go sb’s anger*, *give in to the desire*
nonPerm1Fact	*suppress a laugh*, *check an impulse*, *hold back a tear*
Perm1Manif	*display one’s strength*, *exhibit impatience*, *show tact*
nonPerm1Manif	*conceal a smile*, *hide one’s hatred*, *stifle one’s laughter*
Liqu*i* is from Lat. *liquidare*, liquidate. The value means ‘the *i*th participant of the event does something so that the event ceases to occur or does not occur’. Liqu has the same meaning without reference to a participant of the event.	

“the same grammatical category that share some semantic characteristics in common, but fail to share others.

So, for example, the set of basic color terms forms a paradigm whose members are adjectives (or nouns), each referring to a different section of the color spectrum.

Not all paradigms are semantically defined, of course. Inflectional paradigms, for instance, include the possible variations

Table 4. Syntagmatic lexical functions (continuation of Tables 2 and 3)

Lexical function	**Examples**
LiquOper2	*exempt sb from liability*, *release sb from debts*, *release sb from his/her duties*
Liqu1Func0. The value means ‘put an end to’.	*withdraw support*, *remove the obstacle*, *end the meeting*
LiquFunc1	*the custom is vanishing*, *get better of one’s shyness*
LiquFunc2	*divert one’s attention from sth*
Liqu1Oper1	*break off*/*wean from*/*wean from a habit*
Real*i* from Lat. *realis*, real. The value means ‘the *i*th participant of an event fulfills the requirement intrinsic to the event, or does with the argument of this function what one is supposed to do with it’. The values are fulfillment verbs, Real*i* differs from Fact*i* and LabReal*ij* with respect to its syntax only.	
Real1. The value means ‘use the argument of this function according to its destination, do with regard to the argument what is normally expected of the first participant’.	*drive a car*, *do one’s duty*, *fulfill one’s obligation*, *follow a principle*
Real2. The value means ‘do with regard to A that which is normally expected of second participant‘.	*accept a challenge*, *pass an examination*, *avenge an insult*, *meet a demand*
Fact*i* from Lat. *factum*, fact. The value means ‘the *i*th participant of an event fulfills the requirement intrinsic to the event or does with the argument of this function what one is supposed to do with it’. The values are fulfillment verbs, Fact*i* differs from Real*i* and Labreal*ij* with respect to its syntax only.	
Fact0. The value means ‘the argument of this function fulfills its own requirement’	*one’s hope comes true*, *the movie is on*, *the suspicion is confirmed*
Fact1	*arouse suspicion*, *stir up hope*
Fact2	*fall under suspicion*, *cherish hope*
Labreal*ij* is a hybrid of Labor and Real. The values of Labreal*ij* are fulfillment verbs, this function differs from Fact*i* and Real*i* with respect to its syntax only.	
Labreal12	*cut sth with the saw*, *hold sth in reserve*
Labreal13	*burn with shame*, *waste one’s health*
Involv is from Lat. *involvere*, drag along. The value is a verb meaning ‘involve the argument of this function’, it links the argument (event) and the name of a non-participant which is affected or acted upon by the event.	*the light floods the room*, *the snowstorm caught him in sth*=place, *the smell filled the room*
Manif is from Lat. *manifestare* ‘manifest’. The value means ‘the argument of this function manifests itself or becomes apparent in sb/sth’.	*amazement lurks in his eyes*, *joy explodes in them*, *scorn is dripping from every word*

> of a lexical item in some inflectional category, such as number.

So, a morphological paradigmatic relation exists between child and children. Paradigmatically related words are, to some degree, grammatically substitutable for each other.

For example, blue, black, and any other member of the color paradigm can sensibly and grammatically occur in the phrase a chair. In this way, paradigmatic relations stand in contrast to syntagmatic relations, which are relations between words that go together in a syntactic structure. For example, we can speak of a syntagmatic relation between eat and dinner.”

Concerning syntagmatic relation, it is:

> “also termed combinational relation, which exists typically, though not necessarily, between expressions of different grammatical categories that can be put together in grammatically well-formed combinations or constructions…, say, between nouns and adjectives, verbs and nouns, or verbs and adverbs etc.” [4].

Table 5. Syntagmatic lexical functions (continuation of Tables 2-4)

Lexical function	Examples
Prox is from Lat. *proximare*, approach. The value means ‘be about to do sth, be on the verge of sth’.	
ProxOper1	*be on the edge of despair*, *be on the brink of disaster*, *be on the verge of tears*
ProxFunc0	*thunderstorm brews*
Prepar*i* is from Lat. *praeparare*, prepare. The value means ‘the *i*th participant of the event prepares sb/sth for sth, get sb/sth ready for normal use or functioning’. Prepar has the same meaning without reference to a participant of the event.	
Prepar1Real1	*propose friendship to sb*
Prepar1Real2	*board a plane*
PreparFact0	*fill up the car*, *load a program into a computer*, *lay the table*
Degrad is from Lat. *degradare*, lower, degrade. The value means ‘degrade, become permanently worse or bad’.	*milk goes sour*, *clothes wear off*, *teeth decay*, *sight dims*, *clothes wear off*, *flower wilts*, *beauty withers*
Son is from Lat. *sonare*, sound. The value means ‘emit a characteristic sound’.	*the dog barks*, *banknotes rustle*, *the waterfall roars*, *a bell chimes*, *chalk grates*, *a pig grunts*, *the alarm clock rings*, *a whip snaps*, *an arrow zings*
Obstr*i* is from Lat. *obstruere*, obstruct. The value means that ‘the *i*th participant of an event functions with difficulty’; alphabetical superscripts specify the aspect of obstruction. Obstr has the same meaning without reference to a participant of the event.	
Obstr	*eyes blur*, *negotiations are stalled*, *economy stagnates*
Obstr2	*sb is short of breath, sb stutters*, *stammers*, *mumbles*
Obstr^{stat}; stat means ‘with respect to vertical position’,	*the body crumples*, *his knees give way*
Stop*i* is from Lat. *stuppare*, stop up, plug. The value means ‘the *i*th participant of an event stop functioning’. Stop has the same meaning without reference to a participant of the event.	
Stop	*his voice breaks*, *her heart is stopping*, *her heart broke*
Stop2	*sb loses his/her breath*
Excess*i* is from Lat. *excessus* (past participle of *excédere*), exceed. The value means ‘the *i*th participant of an event functions in an abnormally excessive way’; alphabetical superscripts specify the aspect of excessive functioning. Excess has the same meaning without reference to a participant of the event.	
Excess	*the engine races*, *sweat rolls down across sb’s forehead*
Excess2	*sb has palpitations*
Excess^{motor}; motor means ‘with respect to movements’.	*the eyes pop out on stalks*, *the heart pounds*, *races*
Excess2^{motor}	*sb grinds his teeth*, *sb is bathed in sweat*
Excess^{color} ; color means ‘with respect to color’.	*cheeks glow*
Excess1^{color}	*be red-cheeked*
Excess^{dim}; dim means ‘with respect to dimension/size’.	*the eyes are like saucers in sb’s head*
Excess^{fulg}; fulg means ‘with respect to brightness’.	*eyes flash/glitter*
Excess^{trem}; trem means ‘with respect to trembling’.	*his hands were shaking*
Excess^{t0}; t^{0} means ‘with respect to temperature’.	*her cheeks burnt*

Paradigmatic lexical functions represent paradigmatic relations, and syntagmatic relations are formalized by syntagmatic lexical functions.

In the example in Section 2, S0 is a paradigmatic lexical function, and IncepOper1 is a syntagmatic lexical function.

Mel’čuk [21] defines 27 paradigmatic and 37 syntagmatic lexical functions.

We give examples of paradigmatic lexical functions in Table 1, and in Tables 2-6 we included syntagmatic lexical functions whose values are verbs and whose arguments are nouns. For

Table 6. Syntagmatic lexical functions (continuation of Tables 2-5)

Lexical function	Examples
Sympt is from Lat. *symptom*, a symptom of sth. The value is a verbal expression meaning ‘a symptom of sth’, it denotes a bodily reaction, i.e., a symptom of an emotional or physical state.	
Obstr(1)-Sympt23(2)	1=breath, 2=anger: *sb chokes with anger* 1=speech, 2=anger: *sb sputters with anger*
Stop(1)-Sympt1(2)	1=speech, 2=amazement: *sb is dumbstruck*
Excess^{motor}(1)-Sympt1(2)	1=hair, 2=fear: *sb’s hair stands on end* 1=eyes, 2=amazement: *sb’s eyes start from their sockets*
Excess^{motor}(1)-Sympt231(2)	1=mouth, 2=amazement: *sb opens his/her mouth wide with amazement*
Excess^{motor}(1)-Sympt13(2)	1=mouth, 2=astonishment: *one’s jaw drops in astonishment* 1=mouth, 2=surprise: *the mouth hangs open in surprise*
Anti is from Lat. *antonymum*, antonym. The value is negation of an internal element of the argument of this function.	
AntiReal1	*win over one’s cancer*
AntiReal2	*fail an examination*, *reject a piece of advice*, *turn down an application*
Result is from Lat. *resultare*, result. The value means ‘be the expected result of sth’.	
ResultOper3	*have the proposal*

explanations and examples in these tables, we used [2, 6-8, 11, 14, 17, 19, 20-22, 29, 32].

It can be observed in Tables 2-6 that some lexical functions represent one simple meaning or a single semantic unit, such functions are called simple (Oper, Func, Labor).

There are lexical functions that formalize combinations of unitary meanings, they are called complex lexical functions (IncepOper, CausFunc). The two categories of simple and complex lexical functions constitute the second typology used to describe and classify lexical functions.

The third typology of lexical functions, as we mentioned in the beginning of this section, considers if lexical functions are applicable to broad or narrow word classes: here standard and non-standard lexical functions are distinguished.

Lexical functions in Tables 1-2 are standard; an example of a non-standard lexical function is *black coffee*, where *black* is used in the meaning ‘without dairy product’, and such meaning is applicable only to the narrow class of beverages [21].

4 Detection of Lexical Functions

Lexical functions represent common semantic and syntactic patterns of certain word classes and can aid in many tasks of natural language processing, lexical and syntactic disambiguation being the most fundamental one among them. In this section, we review some research works on automatic detection of lexical functions in texts.

Wanner [34] suggested to regard the task of lexical function automatic detection as classifying collocations according to the lexical function typology and applied the nearest neighbor technique to resolve it. The experiments were made on two groups of Spanish verb-noun collocations: one with emotion nouns and the other with field-independent nouns. The lexical function learning was done using hypernym information from the Spanish part of EuroWordNet [33]. The highest F-measure achieved for field-independent collocations in the experiments was 0.76 for CausFunc0. An average F-measure of about 0.70 was obtained over all experiments.

The same average result was shown on more data in [35] over four learning methods: nearest neighbor, Naïve Bayes, tree-augmented network, and ID3-algorithm. In this work, the best F-measure on field-independent verb-noun collocations (0.76) was achieved by ID3-algorithm.

Archer [1] experimented on extracting collocations of the adjectival/adverbial lexical function Magn (from Lat. *magnus*, big, great. Magn

Table 7. Our best results on detecting lexical functions

Lexical Function	F1	Algorithm
Oper1	0.879	SimpleCart
Func0	0.824	AttributeSelectedClassifier
ContOper1	0.800	LWL
CausFunc1	0.744	RotationForest
Oper2	0.739	SimpleLogistic
IncepOper1	0.732	Prism
CausFunc0	0.744	RotationForest
Real1	0.667	LogitBoost

values are intensifiers, for example: Magn(*rain*) = *heavy*, Magn(*temperature*) = *high*, Magn(*storm*) = *sever*, Magn(*to weep*) = *bitterly*. The research was done on French material. The reported result is 83% of precision. However, the process of extracting Magn collocations was semi-automatic and included human intervention, which improved the results significantly.

Ramos *et al*. [28] proposed an algorithm to retrieve collocations of the type 'support verb + object' from the semantically annotated FrameNet corpus of examples [31]. Their interest was to see whether the collocations they extracted were of Oper*n*. (in this article we denominated this function as Oper*i*).

The authors assumed that certain syntactic, semantic, and collocation annotations in the FrameNet corpus could signal that a particular collocation belonged to Oper*n*. The proposed algorithm was tested on 208 instances and showed an accuracy of 76%.

In our previous research, for automatic detection of lexical functions, we used a dataset of Spanish verb-noun collocations [15] annotated manually with lexical functions and the Spanish WordNet version 2000611 ([33], Spanish WordNet online[1]) senses.

The dataset was submitted to 66 supervised machine learning techniques implemented in Weka 3-6-12-x64[2] to classify each sample in the dataset set as belonging or not to a particular lexical function (binary yes-no classification) using 10-fold cross validation. Table 7 presents the best results for eight lexical functions in terms of F1-measure indicating what algorithm achieved this result.

5 Application of Lexical Functions

Lexical functions possess a number of important properties, which make them an effective tool for natural language processing:

- Lexical functions are universal; it means that a significantly little number of lexical functions (about 70) represent the fundamental semantic relations between words in the vocabulary of any natural language on the one hand, and the basic semantic relations which syntactically connected wordforms can obtain in a text on the other hand.
- Lexical functions are characteristic of idioms in many natural languages and can serve as a typology for classification of idioms, collocations, and other types of restricted lexical co-occurrence.
- Lexical functions can be paraphrased. For example, the lexical functions Oper and Func can form combinations with their arguments, which are synonymous to the basic verb *control* as in the following utterances: *The government controls prices – The government has control of prices* – The *government keeps prices under control – The prices are under the government's control.*

Lexical functions can be used to resolve syntactic ambiguity. In such cases, syntactically identical phrases are characterized by different lexical functions, which serve as a tool for disambiguation. For example, consider two phrases: *support of the parliament* and *support of the president.* In the first phrase *support* is the object, but in the second phrase *support* functions syntactically as the subject and semantically as the agent.

The surface phrase structure in both cases is identical: *support* + *of* + noun; this fact causes syntactic ambiguity and due to it both phrases may have both meanings: 'support given by the

[1] http://adimen.si.ehu.es/cgi-bin/wei/public/wei.consult.perl/

[2] http://www.cs.waikato.ac.nz/ml/weka/

parliament (by the president)’, which syntactically is the subject interpretation with the agentive syntactic relation between *support* and the subordinated noun, and ‘support given to the parliament (to the president)’, which syntactically is the object interpretation with the first completive syntactic relation between *support* and the subordinated noun. This type of ambiguity is often extremely difficult to resolve, even within a broad context. Lexical function’s verbs can be successfully used to disambiguate such phrases because they impose strong limitations on the syntactic behavior of their arguments in texts. Now let us view the same phrases in a broader context.

The first example is *The president spoke in support of the parliament,* where the verb *speak in* is Oper1 of the noun *support,* i.e., Oper1(*support*) = *speak in*. Oper1 represents the pattern ‘Agent performs w_0’ (where w_0 is the argument of Oper1), so *the president* is interpreted as the agent, and *support* as the object. Therefore, *The president spoke in support of the parliament* can only be interpreted as describing the support given to the parliament, with *parliament* having the syntactic function of the complement of *support*.

On the other hand, verbs of Oper2 participate in another pattern: ‘Patient undergoes w_0’. So Oper2 verb is by definition a verb whose grammatical subject represents the patient of w_0 and in the utterance *The president enjoyed* (Oper2) *the support of the parliament,* the phrase *the support of the parliament* implies the support given to the president by the parliament, with *parliament* having the syntactic function of the agentive dependent of the noun *support*.

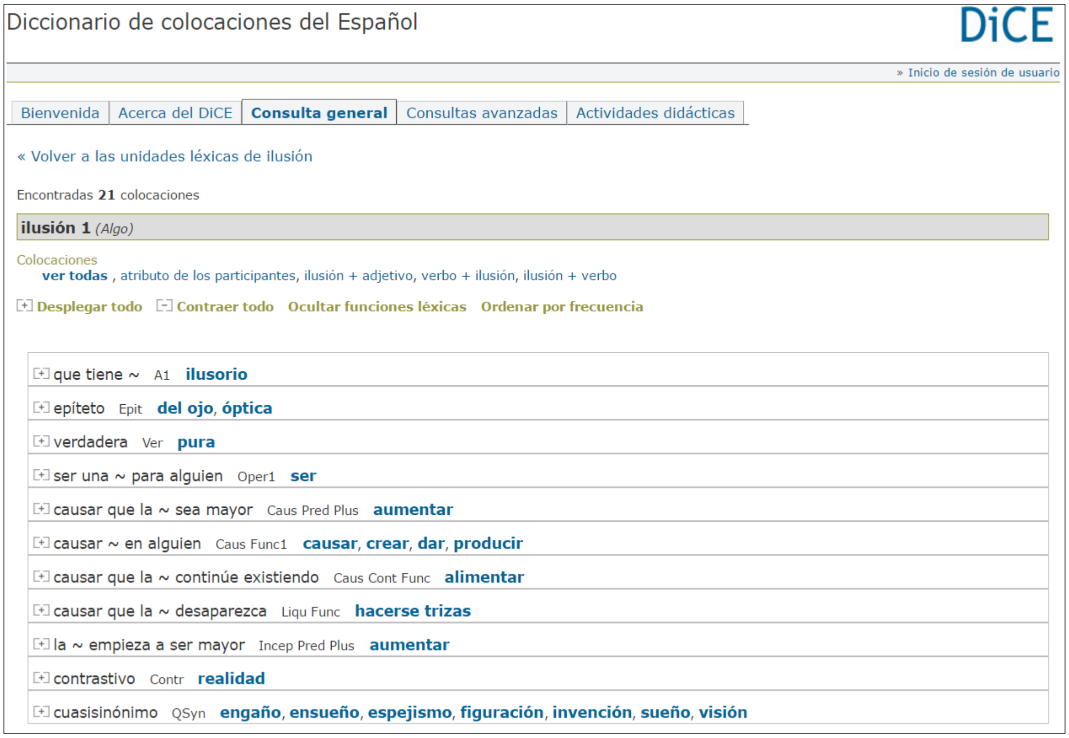
Lexical ambiguity is another issue in natural language processing which lexical functions can help to resolve. For instance, the Russian expression *provodit’ razlichie* and its direct English equivalent *to draw a distinction* can be analyzed as composed of Oper1 + its argument. Taken in isolation, the Russian and the English verbs are extremely polysemous, and choosing the right sense for the given sentence becomes a formidable problem. *Provodit’*, for example, has half a dozen senses ranging from ‘spend’ via ‘perform’ to ‘see off’, while *draw* is a polysemous verb for which dictionaries list 50 or more senses.

However, in both expressions the mutual lexical attraction between the argument of the lexical function and its value is so strong that, once the fact of their co-occurrence is established by the parser, we can safely ignore all other meanings and keep for further processing only the one relevant here (Oper1).

Computer-assisted language learning is another area where lexical functions can be useful. It is a well-known fact in second language teaching practice that collocations are difficult to master by learners, so learner’s speech often sounds unnatural due to errors in restricted lexical co-occurrence. To deal with this issue, a lexical function dictionary can be used whose advantage is that it includes the linguistic material on word combinations which is absent in word dictionaries.

Due to semantic universality and cross-linguistic idiomaticity of lexical functions, they can be employed in machine translation. These characteristics make lexical functions an ideal tool for selecting idiomatic translations of set expressions in a machine translation system. *They took a walk after lunch* is translated into Spanish by Google Translate as *Tomaron un paseo después del almuerzo*. In English, Oper1(*walk*) = *take*, but in Spanish Oper1 of the argument *paseo* (English *walk*) is *dar* (English lit. *give*). So Oper1(*paseo*) = *dar*, however, the system translated the collocation *take a walk* literally as *tomar paseo*, since *take* is literally *tomar* in Spanish. Therefore, a module that annotates word combinations with lexical functions can be included in any machine translation system to improve the quality of translation of collocations and idiomatic expressions.

Probably, the area in which lexical functions are used the most is lexicography: in lexicons and dictionaries of collocations. Automatic semantic analysis of texts with high precision usually requires the use of semantically annotated lexical resources, since the approaches based on word counts in models of distributional semantics do not produce high quality and meaningful interpretation yet. Since lexical functions have the power to generalize and represent combinability of a lexical unit with other lexical units, a lexicon organized with respect to various lexical functions of each entry can serve as a valuable resource for any task of automatic language processing in which semantic procedures are performed on a shallow or deep level.



« Volver a las unidades léxicas de ilusión

Encontradas **21** colocaciones

ilusión 1 *(Algo)*

Colocaciones
ver todas , atributo de los participantes, ilusión + adjetivo, verbo + ilusión, ilusión + verbo

que tiene ~	A1	**ilusorio**
epíteto	Epit	**del ojo**, **óptica**
verdadera	Ver	**pura**
ser una ~ para alguien	Oper1	**ser**
causar que la ~ sea mayor	Caus Pred Plus	**aumentar**
causar ~ en alguien	Caus Func1	**causar**, **crear**, **dar**, **producir**
causar que la ~ continúe existiendo	Caus Cont Func	**alimentar**
causar que la ~ desaparezca	Liqu Func	**hacerse trizas**
la ~ empieza a ser mayor	Incep Pred Plus	**aumentar**
contrastivo	Contr	**realidad**
cuasisinónimo	QSyn	**engaño**, **ensueño**, **espejismo**, **figuración**, **invención**, **sueño**, **visión**

Fig. 1. *Ilusión* entry in DiCE

Syntagmatic lexical functions can serve as a typology of collocations organizing and structuring collocation dictionaries because lexical functions categorize diverse restricted word co-occurrences.

An example of such a collocation dictionary is DiCE, a Spanish collocation dictionary accessible online[3]. The DiCE includes lexical units, that is, word senses (not concepts or synsets as in WordNet[4]) in the domain of sentiment and emotion, and their collocations labeled with lexical functions, see an example entry of *ilusión* (illusion) in Fig. 1.

Speaking of simple lexicons as sets of entries for every word (lexical unit), lexical functions can be associated with entries showing the types of phrases a given word can form with other words. Lareau *et al*. [16] give an example of the entry for the noun *attention*: the entry includes its argument structure and lexical functions (definitions of lexical functions given in this example and not explained in Section 3 can be consulted in [21]):

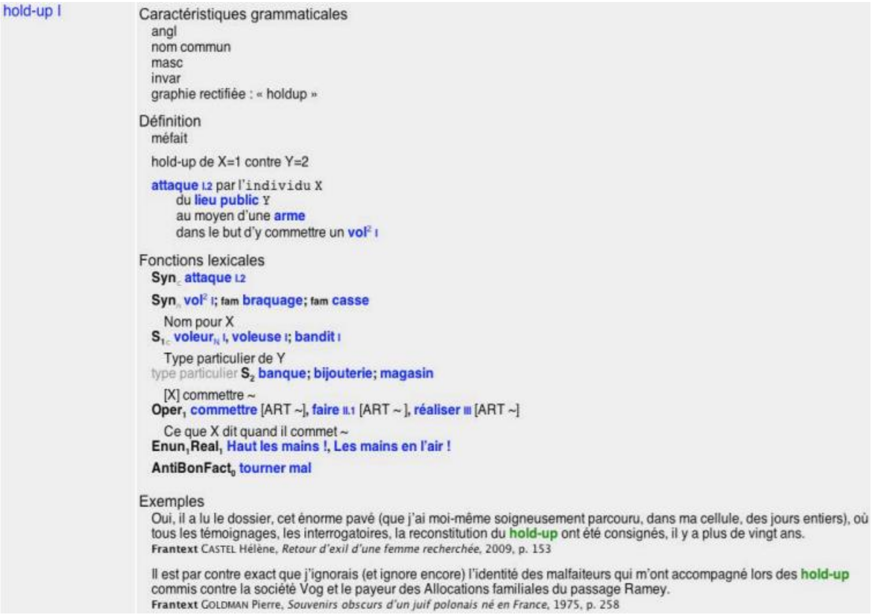
Attention [of X to Y]

Magn	*close/whole/complete/undivided* ~
Func2	X's ~ *is on* Y
nonFunc0	X's ~ *wanders*
Oper12	X *gives his/pays* ~ *to* Y
Oper2	Y *attracts/receives/enjoys* X's ~
Oper2+Magn	Y *is in the center of* ~ (*of many* Xs)
IncepOper12	X *turns his* ~ *to* Y
IncepOper2	Y *gets* X's ~
ContOper2	Y *holds/keeps* X's ~
CausFunc2	Z *draws/calls/brings* X's ~ to Y
LiquFunc2	Z *diverts/distracts/draws* X's ~ *from* Y

A similar kind of a lexicon is presented in [9]. It was compiled manually for French and is called the French Lexical Network (fr-LN). In this resource, each entry represents one sense of a word, i.e., lexical unit, and the entries (vertices of the graph) are connected by arcs labeled with lexical functions. The corresponding English Lexical

[3] http://www.dicesp.com/paginas/

[4] https://wordnet.princeton.edu/



hold-up I

Caractéristiques grammaticales
angl
nom commun
masc
invar
graphie rectifiée : « holdup »

Définition
méfait

hold-up de X=1 contre Y=2

attaque I.2 par l'individu X
du lieu public Y
au moyen d'une arme
dans le but d'y commettre un vol^2 I

Fonctions lexicales
Syn$_{\subset}$ attaque I.2

Syn$_{\cap}$ vol^2 I; fam braquage; fam casse

Nom pour X
S$_{1\subset}$ voleur$_N$ I, voleuse I; bandit I

Type particulier de Y
type particulier **S**$_2$ banque; bijouterie; magasin

[X] commettre ~
Oper$_1$ commettre [ART ~], faire II.1 [ART ~], réaliser III [ART ~]

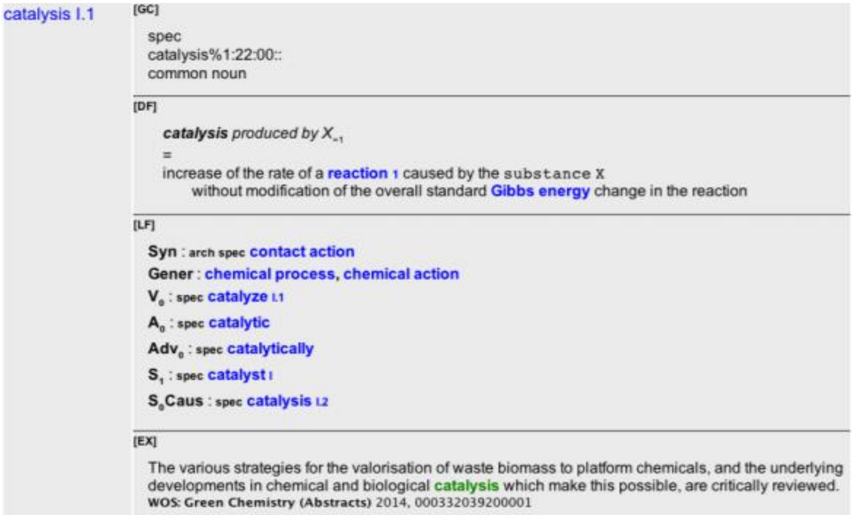
Ce que X dit quand il commet ~
Enun$_1$**Real**$_1$ Haut les mains !, Les mains en l'air !

AntiBonFact$_0$ tourner mal

Exemples
Oui, il a lu le dossier, cet énorme pavé (que j'ai moi-même soigneusement parcouru, dans ma cellule, des jours entiers), où tous les témoignages, les interrogatoires, la reconstitution du hold-up ont été consignés, il y a plus de vingt ans.
Frantext CASTEL Hélène, *Retour d'exil d'une femme recherchée*, 2009, p. 153

Il est par contre exact que j'ignorais (et ignore encore) l'identité des malfaiteurs qui m'ont accompagné lors des hold-up commis contre la société Vog et le payeur des Allocations familiales du passage Ramey.
Frantext GOLDMAN Pierre, *Souvenirs obscurs d'un juif polonais né en France*, 1975, p. 258

Fig. 2. *Hold up* I entry in the French Lexical Network



catalysis I.1

[GC]
spec
catalysis%1:22:00::
common noun

[DF]
catalysis *produced by* $X_{=1}$
=
increase of the rate of a reaction I caused by the substance X
without modification of the overall standard Gibbs energy change in the reaction

[LF]
Syn : arch spec contact action
Gener : chemical process, chemical action
V$_0$: spec catalyze I.1
A$_0$: spec catalytic
Adv$_0$: spec catalytically
S$_1$: spec catalyst I
S$_0$**Caus** : spec catalysis I.2

[EX]
The various strategies for the valorisation of waste biomass to platform chemicals, and the underlying developments in chemical and biological catalysis which make this possible, are critically reviewed.
WOS: Green Chemistry (Abstracts) 2014, 000332039200001

Fig. 3. *Catalysis* I.1 entry in a lexical system of English chemical terms

Network (en-LN) was generated from WordNet 3.0[5] [5].

There are four differences between WordNet and en-LN: first, WordNet is a graph of synsets, and en-LN is a graph of word senses (lexical units); second, the WordNet graph is partitioned based on four parts of speech (noun, verb, adjective, adverb) and the en-LN is not partitioned; third, the WordNet graph is organized hierarchically from top to bottom, and the en-LN has multidimensional organization; forth, WordNet is chiefly based on hypernym-hyponym relation between synsets and the en-LN is based on lexical function relations between lexical units.

[5] https://wordnet.princeton.edu/

Table 8. Examples of collocations and their lexical functions from DiCoInfo

Base of collocation	**Collocate**	**Lexical Function**	**Meaning and syntactic pattern**
programmer	the ~ write…	Fact2	The p. acts on the program
dialog box	open a ~	Real1	The user uses a d.
program	quit a ~	FinReal1	The user stops using a p.
Internet	brows the ~	Real1	The user uses the I.
keyboard	enter … on a ~	Labreal12	The user uses a k. to act on the data
account	access an ~	IncepReal1	The user stars using an a.

Table 9. Entry for *pollute* in DiCoEnviro, see definitions of lexical functions used here in [18]

Base of collocation	**Lexical Function**	**Value of lexical function**
pollute_1b	ResultConv32	pollute_1a
pollute_1b	Anti-2[5]	depollute
pollute_1a	S0	pollution_1b.1
pollute_1a	S*res*	pollution_1b.2
pollute_1b	S1[6]	polluter_1
pollute_1a	S1	pollutant_1
pollute_1b	S3	pollutant_1
pollute_1a	A1	polluting_1
pollute_1b	A3	polluting_1
pollute_1b	QSyn	contaminate
polluting_1	QAnti	green
polluting_1	Anti	clean_1

Gader *et al*. [9] used 12 lexical functions to label lexical relations used in WordNet: seven lexical functions explained in Section 3 ($\text{Syn}_{\cap}$, $\text{Anti}_{\cap}$, Gener, Mult, Sing, A2, Caus), four lexical functions defined in [26] as Cf, Hypo, Holo, Mero, and one non-standard lexical function (Unspecified derivative). Continuing the work in [9], Polguère [27] presents an example of an entry in the French Lexical Network (Fig. 2).

Applying the principles of the fr-LN, Ingrosso and Polguère [12] developed a lexical system of English chemical terms, see an example entry in Fig. 3. Jousse *et al*. [13] designed and compiled the DiCoInfo, Dictionnaire fondamental de l'informatique et de l'Internet, an online specialized database of collocations in the domain of computing and the Internet[6]. The dictionary allows for search of a collocate or collocates of a given word to convey a required meaning in the process of text generation. The dictionary is structured according to syntagmatic lexical functions used for representing collocations, see examples in Tables 2-6.

The authors grouped the original lexical functions of [21] in several larger semantic classes such as 'use', 'create', 'place somewhere', among others. Table 8 gives examples of some entries in the DiCoInfo.

Another example of a lexicon based on lexical functions is DiCoEnviro, a lexicon of words related to the domain of environment described in [18]. Paradigmatic lexical functions of the word *pollute* in its different senses shown in Table 9 give a glimpse of this lexical resource.

[6] http://olst.ling.umontreal.ca/cgi-bin/dicoinfo/search.cgi/

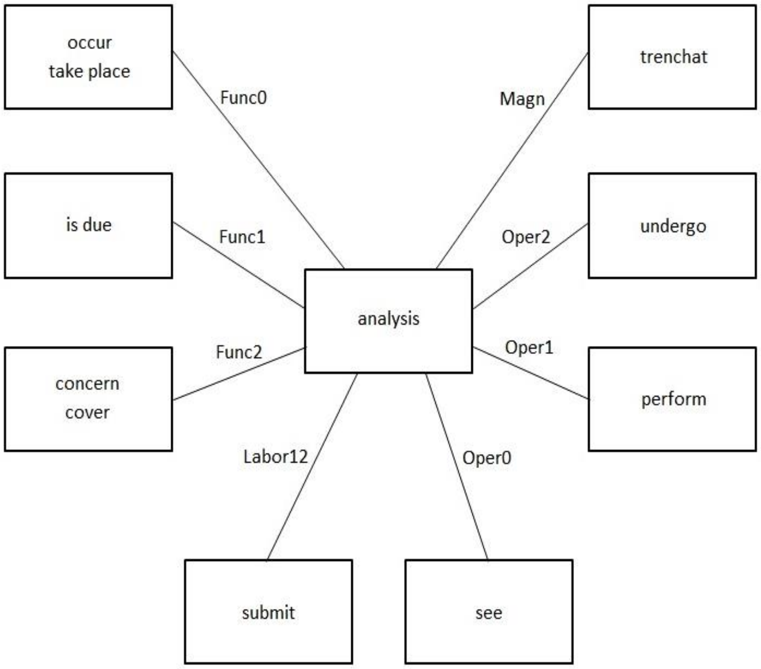


Fig. 4. Semantic network for the noun *analysis*

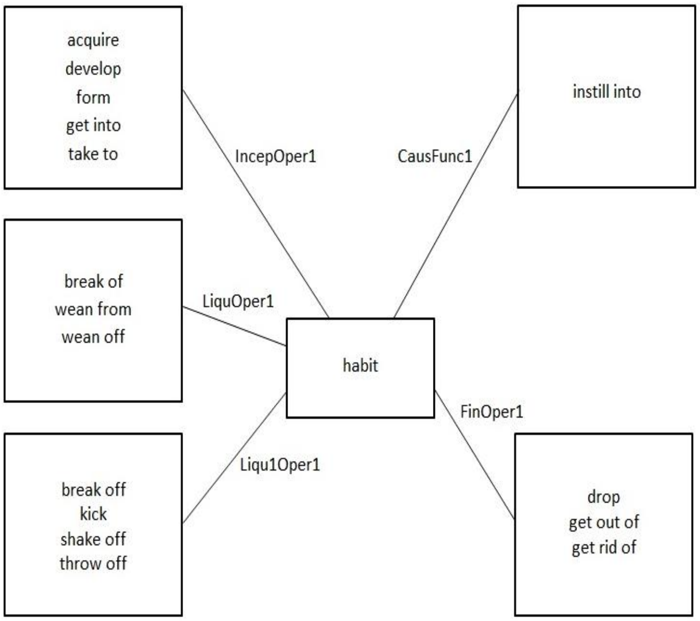


Fig. 5. Semantic network for the noun *habit*

Lexical functions are a convenient tool to apply in interfaces between a lexicon and an ontology [3] as well as in semantic networks [8]. In semantic networks, words are nodes and lexical functions are labels attached to the arcs, see Figs 4, 5 with an example of semantic relations for the nouns *analysis* and *habit*.

Lexical functions can be applied in text generation using semantic and syntactic patterns corresponding to them creating relevant, grammatically correct and naturally sounded word combinations, see examples Table 7. The lexical function patterns can be employed also in other natural language processing tasks: parsing, semantic role tagging, text analysis, etc.

6 Conclusions

In this article, we discussed lexical functions [21], their identification in texts and usage in natural language processing. Lexical function is a formal representation of lexical relations on both paradigmatic and syntagmatic levels.

Paradigmatic lexical functions capture in a systematic way the fundamental relations between word senses (lexical units) such as synonymy and antonymy of various types, hypernymy, hyponymy, derivational relations, among others.

Syntagmatic lexical functions deal with a linguistic phenomenon, which has been an issue in natural language processing for many years, namely, restricted lexical co-occurrence or collocation. These lexical functions categorize collocations into classes depending on their semantic and syntactic features as well as predicate-argument structure, which allows for generalizations in such a diverse part of lexicon as collocations.

Therefore, the most common use of lexical functions is lexicography where words are arranged in semantic networks with lexical functions are relations among them. However, lexical functions serve as an effective instrument in resolving other issues: lexical and syntactic ambiguity, machine translation, second language acquisition, periphrasis. In future, we believe that lexical functions will find application on many other language tasks related to parsing, semantic role labelling, text analysis, generation and classification, among many others.

Acknowledgments

The work was done under partial support of Mexican Government: SNI, COFAA-IPN, BEIFI-IPN, and SIP-IPN grant 20201948.

References

1. **Archer, V. (2007).** Using conceptual vectors to get Magn collocations (and using contrastive properties to get their translations). **Gerdes, K., Reuther, T., Wanner, L. (Eds.)** *Proceedings of the Third International Conference on Meaning-Text Theory*, pp. 57–65, München-Wien: Wiener Slawistischer Almanach, Sonderband 69.
2. **Apresjan, J.D. (1995).** *Leksičeskaia Semantica* (*Lexical Semantics*, in Russian), Vol. 1, Moscow: Vostochnaya Literatura RAN.
3. **Boguslavsky, I., Iomdin, L., Sizov, V., & Timoshenko, S. (2010).** Interfacing the lexicon and the ontology in a semantic analyzer. *23rd International Conference on Computational Linguistics*, pp. 67–76.
4. **Cheng, X. (2016).** A study on lexical sense relations from the perspective of vocabulary breadth and word frequency. *Theory and Practice in Language Studies*, Vol. 6, No. 5, pp. 988–993.
5. **Fellbaum, C. (1998).** *WordNet: An Electronic Lexical Database*, Cambridge, MA: MIT Press.
6. **Fontenelle, T. (1994).** Using lexical functions to discover metaphors. *Proceedings of the 6th EURALEX International Congress*, pp. 271–278.
7. **Fontenelle, T. (1996).** Ergativity, collocations and lexical functions. *Euralex´96 Proceedings*, pp. 209– 222.
8. **Fontenelle, T. (1997).** Using a bilingual dictionary to create semantic networks. *International Journal of Lexicography*, Vol. 10, No. 4, pp. 275–303.
9. **Gader, N., Ollinger, S., & Polguère, A. (2014).** One lexicon, two structures: So what gives? *Seventh Global Wordnet Conference*, pp. 163–171.
10. **Gangemi, A. (2005).** Ontology design patterns for semantic web content. *International Semantic Web Conference*, pp. 262–276.
11. **Heylen, D., Maxwell, K.G., & Verhagen, M. (1994).** Lexical functions and machine translation. *Proceedings of the 15th Conference on Computational Linguistics*, Vol. 2, pp. 1240–1244.

12. **Ingrosso, F. & Polguère, A. (2015).** How terms meet in small-world lexical networks: The case of chemistry terminology. *Terminology and Artificial Intelligence,* pp. 167–171.
13. **Jousse, A.L., L'Homme, M.C., Leroyer, P., & Robichaud, B. (2011).** Presenting collocates in a dictionary of computing and the Internet according to user needs. *Proceedings of the 5th International Conference on Meaning-Text Theory*, pp. 134–144.
14. **Kahane, S. & Polguere, A. (2001).** Formal foundation of lexical functions. *Proceedings of ACL/EACL Workshop on Collocation*, pp. 8–15.
15. **Kolesnikova, O. (2014).** Discriminative ability of WordNet senses on the task of detecting lexical functions in Spanish verb-noun collocations. *International Journal of Computational Linguistics and Applications*, Vol. 5, No. 2, pp. 67–94.
16. **Lareau, F., Dras, M., Börschinger, B., Turpin, M., Butt, M., & King, T. H. (2012).** Implementing lexical functions in XLE. *Proceedings of the Lexical Functional Grammar Conference*, pp. 362–382.
17. **Lemnitzer, L., & Geyken, A. (2015).** Semantic modeling of collocations for lexicographic purposes. *Journal of Cognitive Science*, Vol. 16, No. 3, pp. 200–223.
18. **L'Homme, M. C. (2012).** Using ECL (Explanatory Combinatorial Lexicology) to discover the lexical structure of specialized subject fields. *Words, Meanings and other Interesting Things. A Festschrift in Honour of the 80th Anniversary of Professor Igor Alexandrovic Mel'cuk*, pp. 378–390, Moscow: RCK.
19. **McKeown, K., & Radev, D. (2000).** *Collocations. Handbook of Natural Language Processing*. Marcel Dekker.
20. **Mel'čuk, I.A. (1974).** *Opyt Teorii Lingvističeskix Modelej "Smysl ↔ Tekst"* (*'A Theory of the Meaning-Text Type Linguistic Models'*, in Russian), Moskva: Nauka.
21. **Mel'čuk, I. A. (1996).** Lexical functions: A tool for the description of lexical relations in a lexicon. **In Wanner, L. (Ed.).** Lexical Functions in Lexicography and Natural Language*. Processing,* pp. 37–102, Benjamins Academic Publishers.
22. **Mel'čuk, I.A. (1998).** Collocations and lexical functions. **In Cowie A.P. (Ed.).** *Phraseology: Theory, Analysis, and Applications*, pp. 25–53, Oxford: Clarendon Press.
23. **Mel'čuk, I. A., & Žolkovskij, A. K. (1970).** Towards a functioning 'Meaning-Text' model of language. *Linguistics*, Vol. 8, No. 57, pp. 10–47.
24. **Murphy, M.L. (2003).** *Semantic Relations and the Lexicon: Antonymy, Synonymy and other Paradigms*. Cambridge University Press.
25. **Nivre, J. (2005).** Dependency grammar and dependency parsing. *MSI Report*, Vol. 5133, No. 1959, pp. 1–32.
26. **Polguère, A. (2007).** Lexical function standardness. *Selected Lexical and Grammatical Issues in the Meaning-Text Theory*: *In Honour of Igor Mel'cuk*, Vol. 84, pp. 43–95.
27. **Polguère, A. (2014).** From writing dictionaries to weaving lexical networks. *International Journal of Lexicography*, Vol. 27, No. 4, pp. 396–418.
28. **Ramos, M.A., Rambow, O., & Wanner, L. (2008).** Using semantically annotated corpora to build collocation resources. *Proceedings of LREC*, Marrakesh, Morocco, pp. 1154–1158.
29. **Ramos, M.A. (2015).** Discovering hidden collocations in a bilingual Spanish–English dictionary. **In Kosem, I., Jakubíček, M., Kallas, J., Krek, S. (Eds.).** *Electronic Lexicography in the 21st Century: Linking Lexical Data in the Digital Age*, pp. 170–185.
30. **Roth, M. & Lapata, M. (2016).** *Neural semantic role labeling with dependency path embeddings.* arXiv preprint arXiv:1605.07515.
31. **Ruppenhofer, J., Ellsworth, M., Petruck, M., Johnson, C.R., & Scheffczyk, J. (2006).** *FrameNet II: Extended Theory and Practice*. Berkeley: ICSI.
32. **Song, S.H. (2006).** *Zur Korrespondenz der NV-Kollokationen im Deutschen und Koreanischen*. No. 44, pp. 37–57.
33. **Vossen, P. (1998).** *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*. Dordrecht: Kluwer Academic Publishers.
34. **Wanner, L. (2004).** Towards automatic fine-grained semantic classification of verb-noun collocations. *Natural Language Engineering*, Vol. 10, No. 2, pp. 95–143.
35. **Wanner, L., Bohnet, B., & Giereth, M. (2006).** What is beyond collocations? Insights from machine learning experiments. *Proceedings of the EURALEX Conference*.

Article received on 19/04/2020; accepted on 18/07/2020.
Corresponding author is Olga Kolesnikova.