Revista Signos ISSN 0718-0934
© 2012 PUCV, Chile
DOI: 10.4067/S0718-09342012000100003

45(78) 44-59

Revista Signos
Estudios de Lingüística

Semantic relations between collocations: A Spanish case study

Relaciones semánticas entre las colocaciones: Un estudio de caso del español

Olga Kolesnikova
kolesolga@gmail.com
Centro de Investigación en Computación del
Instituto Politécnico Nacional
México

Alexander Gelbukh
gelbukh@gelbukh.com
Centro de Investigación en Computación del
Instituto Politécnico Nacional
México

Recibido: 24-VIII-2010 / Aceptado: 3-V-2011

Abstract: Linguistics as a scientific study of human language intends to describe and explain it. However, validity of a linguistic theory is difficult to prove due to volatile nature of language as a human convention and impossibility to cover all real-life linguistic data. In spite of these problems, computational techniques and modeling can provide evidence to verify or falsify linguistic theories. As a case study, we conducted a series of computer experiments on a corpus of Spanish verb-noun collocations using machine learning methods, in order to test a linguistic point that collocations in the language do not form an unstructured collection but are language items related via what we call collocational isomorphism, represented by lexical functions of the Meaning-Text Theory. Our experiments allowed us to verify this linguistic statement. Moreover, they suggested that semantic considerations are more important in the definition of the notion of collocation than statistical ones.

Key Words: Collocations, lexical functions, machine learning.

Resumen: La Lingüística, siendo el estudio científico del lenguaje humano, intenta describirlo y explicarlo. Sin embargo, es difícil demostrar la certeza de cualquier teoría lingüística por la naturaleza versátil del lenguaje como una convención humana y también por la imposibilidad de investigar todo lo que se habla y se escribe en la vida real. A pesar de estos problemas, las técnicas y los modelos computacionales pueden proporcionar la evidencia para que las teorías lingüísticas sean comprobadas o refutadas. A través de un estudio de caso, realizamos una serie de experimentos por computador en un corpus de colocaciones de verbo-sustantivo en español usando métodos de aprendizaje de máquina, con el fin de comprobar el hecho lingüístico de que las colocaciones del idioma no conforman un grupo sin estructura sino que son unidades lingüísticas relacionadas por medio de lo que denominamos isomorfismo de colocaciones, representado por las funciones léxicas de la Teoría Significado-Texto. Nuestros experimentos nos permiten verificar esa declaración lingüística. Asimismo, los experimentos sugieren que las consideraciones semánticas son más importantes en la definición de colocaciones que las estadísticas.

Palabras Clave: Colocaciones, funciones léxicas, aprendizaje de máquina.

INTRODUCTION

Computer experiments play a very important role in science today. Simulations on computers have not only increased the demand for accuracy of scientific models, but have helped the researcher to study regions which can not be accessed in experiments or would demand very costly experiments.

In this article, we present computational experiments made on the material of Spanish verb-noun collocations like *seguir el ejemplo*, follow the example, *satisfacer la demanda*, meet the demand, *tomar una decisión,* make a decision. The purpose of the experiments is to test a linguistic statement concerning collocational semantics. In Section 1, we present this linguistic statement in detail, in Section 2, describe the experiments as to the language data and methods used, discuss the experimental results in the light of the linguistic statement made before in Section 3, and in Section 4, derive another important inference on the nature of collocation.

It should be added here that testing a linguistic hypothesis on computer models not only demonstrates validity or rejection of the hypothesis, but also motivates the researcher to search for more profound explanations or to explore new approaches in order to improve computational operation. Thus, starting from one linguistic model, the researcher can evaluate it and then go further, sometimes into neighboring spheres of linguistic reality, in her quest of new solutions, arriving at interesting conclusions. The original intent of our research was to test one linguistic model experimentally. The obtained results produced evidence for verifying this model, but they also made it possible to get more insight into the nature of collocation which has been a controversial issue in linguistics for many years.

1. Case study

Before formulating the linguistic point we are going to test via computer experiments, we will first localize it within the vast realm of linguistics. Our statement is concerned with the concept of collocation, one of contemporary controversial issues in theoretical and applied linguistics. Knowledge of collocation is very important in lexicology (Herbst & Mittmann, 2008), translation (Boonyasaquan, 2006), language acquisition (Handl, 2008), and in various tasks of automated processing of natural language (e.g., in automatic word sense disambiguation: Jin, Sun, Wu & Yu, 2007; in machine translation: Wehrli, Seretan, Nerima & Russo, 2009; in text classification: Williams, 2002, etc.).

1.1. Concept of collocation

Since the linguistic point we have dealt with speaks of collocations, we will first give a definition of collocation. Many definitions have been proposed by linguists, the first one was given in (Firth, 1957) where the author sees collocations of a given word as statements of the habitual or customary places of that word. Here are a few examples of collocations in Spanish taken from (Bolshakov & Miranda-Jiménez, 2004): *prestar atención*, give attention, *presidente del país*, president of the country, *país grande*, large country, *muy bien*, very well. In a collocation, one word dominates over the other and determines its choice (Hausmann, 1984). The dominant word is called the 'base', and the other word whose choice is not free but depends on the base is called the 'collocate'. Thus, in the collocation *prestar atención*, the base is *atención* and the collocate is *prestar*; in *país grande*, the base is *país* and the collocate is *grande*. Semantically, the base is used in its typical meaning while the collocate accepts another meaning, not typical for it but determined by the base.

Now we are going to present our linguistic statement. The terms 'collocational isomorphism' and 'lexical function' are explained in the sections that follow.

1.2. Hypothesis

Collocations are not a stock or a 'bag' of word combinations, where each combination exists as a separate unit with no connection to the others, but they are related via collocational isomorphism represented as lexical functions.

1.3. Collocational isomorphism

Considering collocations of a given natural language (this work was fulfilled on Spanish verb-noun collocations), it can be observed that collocations are not just a 'bag' of word combinations, as a collection of unrelated items where no association could be found, but there are lexical relations among collocations, and in particular, we study the lexical relation which may be called 'collocational isomorphism'. It has some resemblance to synonymy among words which is the relation of semantic identity or similarity. Collocational isomorphism is not a complete equality of the meaning of two or more collocations, but rather a semantic and structural similarity between collocations.

What do we mean by semantic and structural similarity between collocations? For convenience of explanation, we will comment on the structural similarity of collocations first. The latter is not a novelty, and a detailed structural classification of collocations (for English) was elaborated and used to store collocational material in the well-known dictionary of word combinations The BBI Combinatory Dictionary of English (Benson, Benson & Ilson, 1997). However, we will exemplify collocational structures with Spanish data, listing some typical collocates of the noun *alegría*, joy:

verb + noun: *sentir alegría*, to feel joy
adjective + noun: *gran alegría*, great joy
preposition + noun: *con alegría*, with joy
noun + preposition: *la alegría de* (*esa muchacha*), the joy of (this girl).

The above examples are borrowed from the dictionary of Spanish collocations entitled *Diccionario de colocaciones del Español* (Alonso Ramos, 2003), a collection of collocations in which the bases are nouns belonging to the semantic field of emotions. So collocations have structural similarity when they share a common syntactic structure.

We say that two or more collocations are similar semantically if they possess a common semantic content. In Table 1, we present collocations with the same syntactic structure, namely, 'verb + noun'. For these collocations, the meaning is given for us to see what semantic element can be found that is common to all of them.

It may be noted that the meaning of all collocations in Table 1 is generalized as 'do, carry out or realize what is denoted by the noun', in other words, that these collocations are built according to the semantic pattern 'do the noun'. In turn, observing the meaning of the nouns, we see that their semantics can be expressed in general terms as 'action' (*uso*, *abrazo*, *medida*) or 'psychological attribute' (*atención*, *interés*), so the resulting semantic pattern of the collocations in Table 1 is 'do an action / manifest a psychological attribute'. Since these collocations share common semantics and structure, we may say that they are

Table 1. Verb-noun collocations and their meaning.

Spanish collocation	English literal translation	Translation into natural English	Meaning of collocation	English translation
hacer uso *dar un abrazo* *prestar atención* *tener interés* *tomar la medida*	*make use* *give a hug* *lend attention* *have interest* *take measure*	*make use* *give a hug* *pay attention* *take interest* *take action*	*usar* *abrazar* *fijarse* *interesarse* *actuar*	*use* *hug* *pay attention* *be interested* *act*

Table 2. Verb-noun collocations grouped according to their common semantic pattern.

Semantic pattern	Spanish collocations	English literal translation	Translation into natural English
create an entity or process	*escribir un libro* *elaborar un plan* *construir la sociedad* *dar vida*	write a book elaborate a plan construct a society give life	write a book develop a plan build a society give life
intensify a property or attribute	*aumentar el riesgo* *elevar el nivel* *desarrollar la capacidad* *mejorar la condición*	increase the risk lift the level develop a capacity improve a condition	increase the risk raise the level develop a capacity improve a condition
reduce a property or attribute	*disminuir la probabilidad* *reducir el consumo* *bajar el precio* *limitar los derechos*	lessen the probability reduce consumption lower the price limit rights	lower chances reduce consumption bring down the price restrict rights
begin to realize an action or begin to manifest an attribute	*iniciar la sesión* *tomar la palabra* *asumir el papel* *adoptar una actitud*	initiate a session take the word assume a role adopt the attitude	start a session take the floor assume a role take the attitude
preserve a property or process	*mantener el equilibrio* *guardar silencio* *seguir el modelo* *llevar una vida*	maintain the balance keep silence follow a model carry a life	keep the balance keep quiet follow an example lead a life

isomorphic, or that they are tied to one another by the relation we termed above as 'collocational isomorphism'. Table 2 gives more examples of isomorphic collocations.

1.4. Collocational isomorphism represented as lexical functions

Several attempts to conceptualize and formalize semantic similarity of collocations have been made. As far back as in 1934, the German linguist Porzig (1934) claimed that on the syntagmatic level, the choice of words is governed not only by grammatical rules, but by lexical compatibility, and observed semantic similarity between such word pairs as dog – bark, hand – grasp, food – eat, cloths – wear. The common semantic content in these pairs is 'typical action of an object'. Research of Firth (1957) drew linguists' attention to the issue of collocation and since then collocational relation has been studied systematically. In the article of Flavell and Flavell (1959) and in the paper by Weinreich (1969), there were identified the following meanings underlying collocational isomorphism: an object and its typical attribute (lemon – sour), an action and its performer (dog – bark), an action and its object (floor – clean), an action and its instrument (axe – chop), an action and its location (sit – chair, lie – bed), an action and its causation (have – give, see – show), etc. Examples from the above mentioned writings of Porzig (1934), Flavell and Flavell (1959), Weinreich (1969) are borrowed from (Apresjan, 1995).

The next step in developing a formalism representing semantic relations between the base and the collocate as well as semantic and structural similarity between collocations was done by Mel'čuk. Up to now, his endeavor has remained the most fundamental and theoretically well-grounded attempt to systematize collocational knowledge. This scholar proposed a linguistic theory called the Meaning-Text Theory, which explained how meaning, or semantic representation, is encoded and transformed into spoken or written texts (Mel'čuk, 1974). His theory postulates that collocations are produced by a mechanism called lexical function. Lexical function is a mapping from the base to the collocate; it is a semantically marked correspondence that governs the choice of the collocate for a particular base. The following definition of lexical function is given in (Mel'čuk, 1996: 40):

> "The term **function** is used in the mathematical sense: $f(X) = Y$. …Formally, a Lexical Function f is a function that associates with a given lexical expression L, which is the argument, or keyword, of f, a set $\{L_i\}$ of lexical expressions – the value

> of *f* – that express, contingent on L, a specific meaning associated with *f*:
>
> $$f(L) = \{L_i\}.$$
>
> Substantively, a Lexical Function is, roughly speaking, a special meaning (or semantico-syntactic role) such that its expression is not independent (in contrast to all "normal" meanings), but depends on the lexical unit to which this meaning applies. The core idea of Lexical Functions is thus lexically bound lexical expression of some meanings."

About 70 lexical functions have been identified in (Mel'čuk, 1996); each is associated with a particular meaning according to which it receives its name. The name of a lexical function is an abbreviated Latin word whose semantic content is closest to the meanings of this lexical function. Using the above notation, the collocation *dar un paseo*, lit. give a walk, is represented as Oper_1(*paseo*) = *dar* where 'Oper' is from Latin *operari* (do, carry out); the argument, or the keyword of this lexical function is *paseo*; its value is *dar*; the subscript 1 stores information concerning the syntactical structure of utterances where the keyword of Oper_1 (*paseo*) is used together with its value (*dar*) and where the first argument of *paseo* (Agent) is lexicalized in speech as the grammatical subject: *Mi abuela* (Agent) *da un paseo por este parque cada sábado*, My grandma takes a walk in this park every Saturday. Other collocations that are isomorphic to *dar un paseo* can be represented likewise, and, in fact, they are the collocations we put in Table 1: *hacer uso* is represented as Oper_1(*uso*) = *hacer*, *dar un abrazo*, as Oper_1(*abrazo*) = *dar*, *prestar atención*, as Oper_1(*atención*) = *prestar*, etc. Another example of a lexical function is Func_0, from Lat. *functionare*, function. The keyword of Func_0 can be an action, activity, state, property, relation, the value of Func_0 has the meaning 'happen, take place, realize itself', and the subscript 0 implies that the keyword functions as the grammatical subject in utterances: Func_1(*viento*) = *soplar* (*el viento sopla*, the wind blows), Func_1(*silencio*) = *reinar* (*el silencio reina*, lit. the silence reigns), Func_1(*accidente*) = *ocurrir* (*el accidente ocurre*, the accident happens). The lexical function 'Real_n' (n = 0, 1, 2...), from Lat. *realis*, real, means 'to fulfill the requirement of the keyword', 'to do with the keyword what you are supposed to with it', or 'the keyword fulfils its requirement'. In particular, Real_1 has the meaning 'use the keyword according to its destination', 'do with regard to the keyword that which is normally expected of its first participant': *conceder amistad a alguien*, to strike up a friendship with somebody, *dar cariño*, to give a cuddle, *consumirse en los celos*, to be consumed by jealousy. Real_2 means 'do with regard to X that which is normally expected of second participant': *recibir cariño de alguien*, to share a cuddle with somebody, *aprobar el examen*, to pass the exam, *vengar la ofensa*, to take revenge for the offense.

There are cases in which a given lexical function represents one elementary meaning, as Oper, Func, Real, for which we have explained their meanings and listed examples. More functions representing elementary meanings have been discovered: Labor (Lat. *laborare*, to work, toil), Incep (Lat. *incipere*, to begin), Cont (Lat. *continuare*, to continue), Fin (Lat. *finire*, to cease), Caus (Lat. *causare*, to cause), Perm (Lat. *permittere*, to permit), Liqu (Lat. *liquidare*, to liquidate), etc. But there are still more cases when the verb's semantic content in verb-noun collocations is complex and includes several elementary meanings. For example, consider the semantics of 'begin to realize an action or begin to manifest an attribute' from Table 2, which consists of two elements: 'begin' and 'realize / manifest'. To represent such compound meanings, complex lexical functions are used, those being combinations of elementary lexical functions, termed 'simple lexical functions'. All lexical functions exemplified above are simple.

It can be noted that the collocational semantics in Table 2 are complex lexical functions. Table 3 presents the meanings from Table 2 through the instrumentality of the lexical function formalism. The meanings are accompanied by sample collocations taken from Table 2 as well. The notation includes the names of lexical functions and the syntactic information concerning grammatical functions of lexicalized semantic roles encoded in subscripts. For the sake of preserving the complete notation, we use the name of the function and the subscripts, but since we are interested in the semantic aspect of collocations, here we leave the subscripts unexplained. However, a detailed description of subscripts and their meanings can be obtained from (Mel'čuk, 1996). Complex lexical functions in Table 3 include the following simple lexical functions as their constituents: Caus, Func, Plus, Minus, Incep, Cont, Oper. All of them, except for Plus and Minus, were introduced earlier in this section. Plus (more) and Minus (less) are self-explanatory.

As we have mentioned, about 70 lexical functions were distinguished in (Mel'čuk, 1996). The only other existing typology based on semantic and syntactic features includes only 15 different types

Table 3. Semantic patterns represented as lexical functions.

Semantic pattern and examples	Complex lexical function representation	Complex lexical function description
create an entity or process *escribir un libro* *dar vida*	CausFunc$_0$(*libro*) = *excribir* CausFunc$_0$(*vida*) = *dar*	CausFunc$_0$ = cause an entity or process to function.
intensify a property or attribute *aumentar el riesgo* *elevar el nivel*	CausPlusFunc$_1$(*riesgo*) = *aumentar* CausPlusFunc$_1$(*nivel*) = *elevar*	CausPlusFunc$_1$ = cause that a property or attribute manifest itself to a larger degree.
reduce a property or attribute *disminuir la probabilidad* *reducir el consumo*	CausMinusFunc$_1$(*probabilidad*) = *disminuir* CausMinusFunc$_1$(*consumo*) = *reducir*	CausMinusFunc$_1$ = cause that a property or attribute manifest itself to a lesser degree.
begin to realize an action or begin to manifest an attribute *iniciar la sesión* *adoptar una actitud*	IncepOper$_1$(*sesión*) = *iniciar* IncepOper$_1$(*actitud*) = *adoptar*	IncepOper$_1$ = cause that an action begin to be realized or an attribute begin to manifest itself.
preserve a property or process *mantener el equilibrio* *guardar silencio*	ContOper$_1$(*probabilidad*) = *disminuir* ContOper$_1$(*probabilidad*) = *disminuir*	ContOper$_1$ = cause that an action continue to be realized or an attribute continue to manifest itself.

```
Algorithm: constructing data sets
Input: a list of 900 Spanish verb-noun collocations annotated with 8 lexical functions
Output: 8 data sets – one for each lexical function
For each lexical function
Create an empty data set and assign it the name of the lexical function.
For each collocation in the list of verb-noun collocations
        Retrieve all hyperonyms of the noun.
        Retrieve all hyperonyms of the verb.
        Make a set of hyperonyms:
        {noun, all hyperonyms of the noun, verb, all hyperonyms of the verb}.
        If a given collocation belongs to this lexical function
            assign '1' to the set of hyperonyms,
        Else   assign '0' to the set of hyperonyms.
        Add the set of hyperonyms to the data set.
Return the data set.
```

Figure 1. Algorithm of compiling the data sets.

of collocations (Benson et al., 1997). Therefore, lexical functions can serve as a more detailed representation of collocational isomorphism.

Now we are going to see if computer experiments can supply evidence to the existence of collocational isomorphism as defined by lexical functions. The idea is to submit a list of collocations to the computer and see if it is able to distinguish collocations belonging to different lexical functions. If a machine can recognize lexical functions, then it is a strong testimony to their existence.

2. Computer experiments

2.1. Outline of the experimental procedure

This section gives an overview of the experiments, and how the data analysis was accomplished. Basically, the experiments consist in asking the computer if a given collocation belongs to a particular lexical function or not. For example, the computer has to decide if *iniciar la sesión* from Table 3 is IncepOper$_1$ or not. The computer's decision is made after the data set analysis. For the experiments, eight lexical functions were chosen. This choice was made on the basis of data available to us. This is further explained in Section 2.2.

For each of eight lexical functions, a data set is compiled according to the algorithm presented in Figure 1. The input to this algorithm is a list of 900 Spanish verb-noun collocations annotated with eight lexical functions. The output is eight data sets, one for each of the selected lexical functions. The data sets include all hyperonyms[1] of each verb and all hyperonyms of each noun in the collocations. The hyperonyms are retrieved from the Spanish WordNet, an electronic dictionary. More details about the list of collocations and the data sets are given in Section 2.2.

The next stage of the experimental procedure is to submit the data sets to machine learning techniques which construct models for making decisions. How the data is analyzed and how the model is build is specific to every machine learning method. The models are evaluated so that one can see whether a method is precise enough on detecting lexical functions. The evaluation is done in terms of F-measure. The methodology is further explained in Section 2.3, and the experimental results are given in Section 2.5.

2.2. Data

In this section, more explanation and details are given as to what data was used and how data sets for machine learning experiments were compiled. Experiments were fulfilled on the material of Spanish verb-noun collocations like *formar un grupo*, form a group, *dar una conferencia*, give a lecture, *destacar la importancia*, emphasize the importance, *presentar información*, present information. 900 verb-noun collocations were extracted from the Spanish Web Corpus automatically using the Sketch Engine, software for automatic text processing (Kilgarriff, Rychly, Smrz & Tugwell, 2004). The Spanish Web Corpus[2] contains 116 900 060 tokens[3] and is compiled of texts found in the Internet. The texts are not limited to a particular topic but touch on any theme which can be discussed on the World Wide Web. We extracted collocations which are most frequently met in the Spanish Web Corpus; therefore, these are most common collocations in contemporary Spanish Internet communication.

The list of 900 verb-noun collocations was manually annotated with lexical functions[4]. Some verb-noun pairs in the list did not have collocational nature, so they were tagged as 'free word combinations'. For example, *poner un ejemplo*, give an example (CausFunc$_0$), *tener un efecto*, have an effect, (Oper$_1$), *dar un salto*, make a leap (Oper$_1$) are collocations characterized by lexical functions and *dar una cosa*, *tener casa*, *dar la mano* are free word combinations. Among 900 most frequent verb-noun collocations, there were 261 free word combinations and 639 collocations belonging to 36 lexical functions.

As it was said in Section 1.4, the overall number of lexical functions that have been identified is 70. This number includes lexical functions found in collocations of various structures: noun-noun, adjective-noun, verb-noun, verb-adverb, etc. In this work, we study only Spanish verb-noun collocations, and we were interested in lexical functions encountered in most frequent of them. We have found out that the list of 900 verb-noun collocations described above contains 36 lexical functions. However, only eight lexical functions of these 36 have the number of collocations sufficient for computer experiments, so they were selected for machine learning experiments. The chosen lexical functions are shown in Table 4, and the number of collocations for each lexical function is presented in Table 5.

The next step in data preparation was to find out in what sense words were used in collocations. So every noun and every verb in the list was disambiguated manually with word senses of the Spanish WordNet (Vossen, 1998). Word senses in this dictionary are designated by numbers and represented by synsets, or synonym sets, consisting of words synonymous with each other and naming one concept. A synset may be accompanied by a brief definition, or 'gloss'. Below we give all senses for the word *broma*, joke, found in the Spanish WordNet; each sense has its number, synset and gloss, words in synsets are written in the form '*word*_number of the sense':

> Sense 1: *broma*_1 *jocosidad*_1 *chanza*_1 *ocurrencia*_1 *gracia*_1 *chiste*_1 a humorous anecdote or remark
> Sense 2: *broma*_2 *vacilada*_1 *burla*_4 a ludicrous or grotesque act done for fun and amusement
> Sense 3: *broma*_3 *jocosidad*_3 activity characterized by good humor
> Sense 4: *broma*_4 *teredo*_1 typical shipworm

After word sense disambiguation was accomplished, we extracted hyperonyms for all words in collocations. Hyperonyms were taken from the same dictionary, i.e., the Spanish WordNet. The purpose was to represent the meaning of each verb-noun collocation by all hyperonyms of the verb and all hyperonyms of the noun. As an example, let us

Table 4. Lexical functions chosen for the experiments.

Lexical function	Meaning	Spanish example	English translation
Oper$_1$	carry out the noun	*alcanzar un objetivo* *satisfacer una necesidad*	realize a goal satisfy a need
Oper$_2$	undergo the noun	*aprobar el examen* *sufrir un cambio*	pass the exam undergo a change
IncepOper$_1$	begin to carry out the noun	*adoptar una actitud* *tomar posición*	adopt an attitude obtain a position
ContOper$_1$	continue to carry out the noun	*guardar silencio* *mantener el equilibrio*	keep silence maintain one's balance
Func$_0$	the noun occurs	*el tiempo pasa* *la razón existe*	time flies the reason exists
CausFunc$_0$	the agent of the noun cause the noun to occur	*establecer un sistema* *producir un efecto*	establish a system produce an effect
CausFunc$_1$	a participant different from the agent of the noun cause the noun to occur	*abrir camino* *producir un cambio*	open the way produce a change
Real$_1$	to fulfill the requirement of the noun	*cumplir el requisito* *solucionar un problema*	fulfill the requirement solve a problem

Table 5. Probability of selecting 'yes' class at random.

Lexical function	Number of examples	Probability of the class 'yes'
Oper$_1$	280	0.311
IncepOper$_1$	25	0.028
ContOper$_1$	16	0.018
Oper$_2$	30	0.033
Real$_1$	61	0.068
Func$_0$	25	0.028
CausFunc$_0$	112	0.124
CausFunc$_1$	90	0.100

take *hacer una broma*, make a joke. We clarify this collocation and get *hacer*_15 *broma*_1. In Fig. 2 we list all hyperonyms of *hacer*_15 and *broma*_1. The words of the collocation, i.e. *hacer*_15 and *broma*_1, are considered hyperonyms of themselves, or zero-level hyperonyms, and are included in the hyperonym set.

Thus the meaning of *hacer una broma* is represented as the hyperonym set {*hacer*_15, *efectuar*_1 *realizar*_6 *llevar_a_cabo*_5 *hacer*_15, *actuar*_2 *llevar_a_cabo*_3 *hacer*_8, *broma*_1 *jocosidad*_1 *chanza*_1 *ocurrencia*_1 *gracia*_1 *chiste*_1, *humorada*_1 *jocosidad*_2, *contenido*_2 *mensaje*_1, *comunicación*_2, *relación_social*_1, *relación*_4, *abstracción*_6} and is supplied to the computer. All 900 collocations are represented likewise and become input data for machine learning techniques. We believe that hyperonym sets have the power of distinguishing collocations belonging to different lexical functions. Therefore, hyperonyms can be a sufficient semantic description of collocations for our purpose.

2.3. Methodology

The task of the computer is to look through collocations of a given lexical function, for example, Oper$_1$, marked as the class 'yes' according to the

Hyperonyms of *hacer_15*
efectuar_1 realizar_6 llevar_a_cabo_5 hacer_15
actuar_2 llevar_a_cabo_3 hacer_8

Hyperonyms of *broma_1*
broma_1 jocosidad_1 chanza_1 ocurrencia_1 gracia_1 chiste_1
humorada_1 jocosidad_2
contenido_2 mensaje_1
comunicación_2
relación_social_1
relación_4
abstracción_6

Figure 2. Hyperonym set for *hacer una broma*.

algorithm in Figure 1, compare them to the rest of the input data we prepared, i.e., to the collocations of all other lexical functions and free verb-noun combinations, marked as the class 'no', and to identify what features are characteristic for collocations of Oper$_1$ (in our data, the features are hyperonyms). In other words, the computer must find what features distinguish Oper$_1$ from other lexical functions. This knowledge will be used later, when the computer is given a list of collocations whose lexical functions are unknown to it. Then the computer's task is to examine these collocations and determine which of them belong to Oper$_1$.

A computational technique used for tasks similar to the one we have just described, is called machine learning. In fact, machine learning is a class of methods developed in the area of artificial intelligence. These methods are based on various mathematical or statistical models and applied to extract knowledge from data: find data patterns, build structural description of data items, classify these items. The field of machine learning has its own concepts and terminology, and we are going to introduce some of the terms in the course of this section to make our exposition more exact. Pieces of data examined by machine learning techniques are called instances, or examples. In our case, an example is a set of all hyperonyms for a particular collocation as represented in Figure 2. Examples can be annotated with respective lexical functions called classes in machine learning, or can be left without such annotation.

Machine learning methods operate in two stages. At the first stage, called learning, computer finds what features are associated with each class of examples in the data set where all examples are tagged with respective classes. Such a set is called a training set. The result of this step is a model built by the computer on the training data. The model is a computational description of patterns automatically found in the data. At the second stage, called testing, the computer tests the model constructed at the learning stage by using it for assigning classes to non-annotated examples (called unseen examples) in a data set called a test set. The procedure ends with the model's evaluation done to check how well the computer predicts classes for unseen examples. For the purpose of evaluation, various methods and metrics are applied. In our experiments, we used the method called 10-fold cross-validation and the metrics called F-measure. We will not explain this method and metrics here since it is not our purpose to go into mathematical and computational details, but a more technical-oriented reader may wish to consult (The University of Waikato, 2010a) on these topics. However, we will give a brief interpretation of F-measure later in this section.

For machine learning experiments, we used WEKA version 3-6-2, open-source machine learning software (Hall, Frank, Holmes, Pfahringer, Reutemann & Witten, 2009; The University of Waikato, 2010b). Basically, we planned to study the performance of two techniques: one based on word frequency count and the other, on rules. As it was said earlier in this section, machine learning methods take advantage of various mathematical or statistical models. Basically, they are built on two types of models, in other words, two strategies or approaches. A lot of methods have been elaborated and implemented in WEKA. They vary in details but still remain within the boundaries of either strategy. Now we turn to considering the two approaches.

2.4. Two approaches in machine learning: Word frequency count and rules

The first approach to finding patterns in data is to count how many times each feature, also called attribute, occur in examples of each class. This gives us an estimation of how certain one can be in assigning an unseen example to this or that class. Operating on our linguistic data, methods based on frequency counts, or statistical methods, calculate the probability of collocations in the input data to belong to a given lexical function under study using Bayes' theorem.

$$P(LF \mid H_1, \dots , H_n) = \frac{P(LF)\, P(H_1, \dots , H_n \mid LF)}{P(H_1,\dots,H_n)}$$

where *LF* is any of eight lexical functions in our experiments; H_1, ... , H_n is a set of all hyperonyms for a collocation; $P(LF|H_1, \dots , H_n)$ is the conditional probability of the collocation to belong to a given lexical function if this collocation has the set of hyperonyms H_1, ... , H_n; $P(LF)$ is the lexical function probability; $P(H_1, \dots , H_n|LF)$ is the conditional probability of the set of hyperonyms if the latter belongs to the lexical function, and $P(H_1, \dots , H_n)$ is the probability of the set of hyperonyms.

Many statistical machine learning methods use Bayes' formula together with optimizations and improvements, but all are based on probabilistic knowledge. More details including formulas for calculating probabilities and measures of likelihood can be found in (Witten & Frank, 2005). A limitation of statistical methods is the assumption that all features in data (hyperonyms in our case) are equally important in contributing to the decision of assigning a particular class to an example and also independent of one another. This is a rather simplified view of data, because in many cases data features are not equally important or independent and this is certainly true for linguistic data, especially for such a language phenomenon as hyperonyms. Graphically, hyperonyms form a hierarchic structure called a tree where every hyperonym has its ancestor (except for the hyperonym at the root of the tree) and daughter(s) (except for hyperonyms at the leaves of the tree). Although statistical methods have weak points, in fact, they perform well enough on such linguistic tasks as automatic speech recognition, part-of-speech tagging, word sense disambiguation, machine translation. In this work, we study how statistical methods perform on the task of assigning lexical functions to collocations.

The second approach in machine learning is based on rules. The computer examines the data and looks for rules which can be inferred from it. Rules are conditional statements of the form:

If Hyperonym$_1$ = *word*$_1$ and Hyperonym$_2$ = *word*$_2$
... and Hyperonym$_n$ = *word*$_z$
then the collocation is of LF,

where LF can be any of eight lexical functions chosen for the experiments. Unlike statistical methods based on probabilistic knowledge, rule-based methods acquire and use conceptual knowledge which is more human-readable and easy to understand. A concept in fact is a number of features that describe an abstract idea. The experimental results are reported in Section 2.5 where we will see if rule-based representations of hyperonyms can effectively describe lexical functions and semantic patterns. In other words, we want to determine the presence or absence of what hyperonyms is characteristic for what lexical function and necessary for distinguishing one lexical function from another.

2.5. Experimental results

As it was explained in Section 2.2, we chose eight lexical functions for our experiments. Using a training set where each collocation is represented by means of hyperonyms, we have tested 47 statistical methods and 21 rule-based methods. The success of methods was evaluated with F-measure. It is a measure that takes into account two numbers: first, how many collocations among which computer said to belong to a given lexical actually belong to this function, and second, how many collocations which belong to a given lexical function were assigned this function by the computer. For more details, see (The University of Waikato, 2010a). Values of F-measure lie within the range from 0 to 1. 'Zero' means that the computer failed to complete its task; 'one' means that the task was accomplished with 100% accuracy, so the higher the value of F-measure, the better the computer performance, in our case, the more precise is its recognition of lexical functions.

The training set submitted to WEKA techniques, or classifiers, is built according to the algorithm in Figure 1. In fact, eight training sets, or data sets, one for each lexical function was experimented with. Each data set contains the same 900 verb-noun collocations represented as sets of hyperonyms. The only difference between these data sets is the value of the class variable, which is 'yes' if a collocation belongs to a given lexical function, and 'no' if it does not.

Usually, in machine learning experiments, and those using WEKA in particular, the classifier ZeroR is chosen as the baseline. ZeroR is a trivial classifier that assigns the majority class to all examples. In our experiments, the majority class is always 'no' since the number of negative examples for each lexical function is much bigger than the number of its positive examples. For example, the number of positive examples for Oper$_1$ is 280 and the number of its negative examples is 900 – 280 = 620. The number of positive examples for the rest seven lexical functions is even less than for Oper$_1$ as it is seen from Table 5, so ZeroR has no sense as the baseline.

However, the baseline can be a random choice of a positive or a negative answer to the question 'Is this collocation of this particular lexical function?' In such a case we deal with the probability of a positive and negative response. Since we are interested in only assigning the positive answer to a collocation, we calculate the probability of 'yes' class for eight lexical functions in the experiments according to the formula: probability of 'yes' = 1 / (the number of all examples / the number of positive examples of a given lexical function). These probabilities will be results of a classifier that assigns the class 'yes' to collocations at random. Since we will compare the probabilities of the random choice with the results obtained in our experiments, we present the former as numbers within the range from 0 to 1 in Table 5 as well as in Table 6.

We obtained F-measure values for all methods applied in the experiments, and for each lexical function, we chose top-performing techniques among rule-based methods. Table 6 presents the results of these techniques together with the results demonstrated by the statistical method called Naïve Bayes (Witten & Frank, 2005). Naïve Bayes is widely used in natural language processing and has proved itself to be one of the most effective methods for accomplishing linguistic tasks. As an example, see (Provost, 1999). In our experiments, this method showed the average F-measure of only 0.145 and was not able to detect some lexical functions at all (F-measure value of 0.000 for IncepOper$_1$, ContOper$_1$, Oper$_2$ and Func$_0$), while rule-based methods significantly outperformed Naïve Bayes and reached the average F-measure of 0.759. Table 6 specifies the names of rule-based methods in WEKA implementation; presents their results as well as the number of examples for each lexical function in the training set. To compare our results with the baseline explained above in the previous paragraphs, the probability values of a random selection of the class 'yes' are demonstrated in the same table.

As it is clearly seen, the performance of Naïve Bayes for six of eight lexical functions is even less than the baseline which is rather low, though the average result of this method is a little bigger than the average baseline. On the contrary, the results of rule-based methods are significantly higher than the baseline.

Table 6. Performance of statistical and rule-based methods.

Lexical function	Baseline	F-Measure		
		Statistical method: Naïve Bayes	Rule-based methods	
			Name	Result
Oper$_1$	0.311	0.735	PART	0.877
IncepOper$_1$	0.028	0.000	JRip	0.757
ContOper$_1$	0.018	0.000	Prism	0.813
Oper$_2$	0.033	0.000	Ridor	0.834
Real$_1$	0.068	0.030	NNge	0.635
Func$_0$	0.028	0.000	DecisionTable	0.824
CausFunc$_0$	0.124	0.303	JRip	0.716
CausFunc$_1$	0.100	0.094	JRip	0.729
Average:	**0.089**	**0.145**		**0.759**

3. Discussion: Testing the linguistic statement

The purpose of this work is to provide evidence for the linguistic statement made in Section 1.2. Now let us review it in the light of our experimental results. The statement affirms that collocations are not a stock, or a 'bag' of word combinations, where each combination exists as a separate unit with no connection to others, but they are related via collocational isomorphism represented as lexical functions.

What evidence have we obtained concerning lexical functions? We presented a sufficient number of collocations annotated with lexical functions to the computer that learned characteristic features of each function. It was demonstrated that the computer was able to assign lexical functions to unseen collocations with a significant average accuracy of 0.759. Is it satisfactory? We can compare our result with computer performance on another task of natural language processing: word sense disambiguation, i.e., identifying the intended meanings of words in context. Today, automated disambiguating systems reach the accuracy of about 0.700 and this is considered a substantial achievement. As an example of such works see (Zhong & Tou Ng, 2010). Therefore, our result is weighty enough to be a trustworthy evidence for the linguistic statement under discussion.

In the Introduction we stated, that if we develop a computer program on the premise of a certain linguistic model and this program accomplishes its task successfully, then the linguistic model being the basis of the program is thus verified. In our experiments, we have observed that machine learning methods are able to detect lexical functions of collocations. Thus lexical functions as a linguistic concept get evidence received in computational experiments which can be repeated on the same data as well as on new data. It means that the formalism of lexical functions is a legitimate model of collocational isomorphism described in Section 1.3.1.

4. Nature of collocation: Statistical vs. semantic approach

What knowledge is necessary and sufficient for the computer to analyze and generate texts in natural language? And what type of knowledge should it be? Up to now, the two foremost approaches in natural language processing have been the statistical and the symbolic.

We touched upon the statistical and symbolic computational strategies in Section 2.4. where methods based on word frequency count (statistical approach) and rule-based methods (symbolic approach) were discussed. It was demonstrated in Section 2.5. that rule-based methods outperformed statistical methods in detecting lexical functions. It means that collocations are analyzed better by rules than by frequency counts; that rules tell us more of what collocations are than frequency counts do; that collocations can be recognized better semantically than statistically.

The fact that the semantic aspect of collocation outweighs the statistical one has an important effect on the definition of collocations. Definition of a concept must contain necessary and sufficient criteria for distinguishing this concept from other concepts. The debate over the most relevant criterion for defining collocations has already lasted over a long period. Should this criterion be statistical or semantic? (Wanner, 2004) gives a good concise overview of this debate. The statistical definition of collocation, i.e. based on probabilistic knowledge, says that collocation is the syntagmatic association of lexical items, quantifiable, textually, as the probability that there will occur, at *n* removes (a distance of *n* lexical items) from an item *x*, the items a, b, c ... (Halliday, 1961). The semantic definition of collocation explains how the collocational meaning is formed: a collocation is a combination of two words in which the semantics of the base is autonomous from the combination it appears in, and where the collocate adds semantic features to the semantics of the base (Mel'čuk, 1995). For example, in the phrase 'She fell to the floor', all the words are used in their typical sense and the verb 'to fall' means 'to drop oneself to a lower position', but when it is said 'She fell in love', we understand that the same verb is not used in its typical, full meaning, but attains a different sense 'begin to experience something'. WordReference Online Dictionary[5] gives a description of this sense: pass suddenly and passively into a state of body or mind. To illustrate the definition, the dictionary provides the following examples: 'to fall into a trap', 'She fell ill', 'They fell out of favor', 'to fall in love', 'to fall asleep', 'to fall prey to an imposter', 'fall into a strange way of thinking'. This meaning of 'fall' is more abstract as compared with its typical meaning given in (WordNet, 2005) 'descend in free fall under the influence of gravity', e.g., 'The branch fell from the tree'. Fall reveals its characteristic meaning in free word combinations, and its more abstract sense, in collocations. What do we mean by more abstract sense? An abstract sense is not independent, it is not complete, but rather can be called a 'semantic particle' whose function is not to express the full semantics, but to add semantic features to the base of collocation.

To explain what is meant by 'adding semantic features to the base', let us make an analogy with semantics of grammatical categories which is also very abstract. The verb be in its function as an auxiliary verb does not express any meaning except abstract grammatical categories of time, aspect, and person. In the sentence 'This castle was built in the 15th century', the verb build carries the meaning of an action, and what be does is adding semantic features to the verb, i.e. that this action took place in past, it is passive, not active, and was applied to a single object, because the grammatical number of 'be' is singular. Likewise, fall does not express an event, or a state, but to the word denoting an event or state 'adds' the semantic feature 'begin to occur'.

According to the semantic definition of collocation, the latter differs from free word combinations in the way it constructs its semantics. While the semantics of a free word combination is the sum of the meanings of its elements, collocational meaning is formed by adding more abstract semantic features expressed by the collocate to the full meaning of the base.

Our experiments showed that collocations are recognized better using rules, or conceptual knowledge. It means that the basic criterion for distinguishing collocations from free word combinations is semantic, so there is a good evidence and reason to build definition of collocation on the semantic, not statistical, criterion.

CONCLUSIONS

It has been demonstrated that computer experiments we made on Spanish verb-noun collocations verify the linguistic hypothesis that collocations are not a random stock of word combinations, but they are semantically and syntactically related to one another. We have found than collocations of the same syntactic structure, namely, verb + noun, are organized in groups with similar semantics. Their similarity is represented by the formalism of lexical functions (Mel'čuk, 1996).

We experimented with 20 statistical and 21 rule-based machine learning techniques on the training set of Spanish verb-noun collocations annotated with eight lexical functions. The obtained results have showed that rule-based methods significantly outperform statistical methods. In particular, we compared the results of the best rule-based methods for detecting lexical functions with the results Naïve Bayes, one of the most efficient methods in natural language processing, on the same task. The average F-measure reached by Naïve Bayes is 0.145 while the average F-measure of rule-based methods is 0.759. This proves that rules capture significant semantic features of collocations which are sufficient for discerning collocational meaning represented by lexical functions.

Statistical machine learning methods use models built on probabilistic knowledge while rule-based methods take advantage of conceptual knowledge. Concepts are semantic units and rules are a means of identifying concepts. Therefore, a better performance of rule-based methods over statistical methods demonstrates that the semantic approach to collocation is more helpful in exploring the nature of such a linguistic phenomenon as collocations.

REFERENCES

Alonso Ramos, M. (2003). Hacia un Diccionario de colocaciones del español y su codificación. In M. A. Martí (Eds.), *Lexicografía computacional y semántica* (pp. 11-34). Barcelona: Edicions de l'Universitat de Barcelona.

Apresjan, J. (1995). *Lexical semantics*. (In Russian). Moscow: Vostochnaya Literatura RAN.

Benson, M., Benson, E. & Ilson, R. (1997). *The BBI Combinatory Dictionary of English: A Guide to Word Combinations*. Amsterdam, Philadelphia: John Benjamins.

Bolshakov, I. A. & Miranda-Jiménez S. (2004). A small system storing Spanish collocations. In A. Gelbukh (Ed.), *Lecture notes in computer science: Computational linguistics and intelligent text processing* (pp. 248–252). Berlin: Springer-Verlag.

Boonyasaquan, S. (2006). An analysis of collocational violations in translation. *Journal of Humanities*, 27, 79–91.

Firth, J. R. (1957). Modes of meaning. In J. R. Firth (Ed.), *Papers in Linguistics 1934–1951* (pp. 190–215). Oxford: Oxford University Press.

Flavell, J. H. & Flavell, E. R. (1959). One determinant of judged semantic and associative connection between words. *Journal of Experimental Psychology*, *58*(2), 159–165.

Handl, S. (2008). Essential collocations for learners of English: The role of collocational direction and weight. In F. Meunier & S. Granger (Eds.), *Phraseology in foreign language learning and teaching* (pp. 43–66). Amsterdam: John Benjamins.

Hausmann, F. J. (1984). Wortschatzlernen ist Kollokationslernen. Zum Lehren und Lernen französischer Wortverbindungen. *Praxis des neusprachlichen Unterrichts*, 31, 395–406.

Hausser, R. (2001). *Foundations of computational linguistics*. Berlin: Springer-Verlag.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. & Witten I. H. (2009). The WEKA Data Mining Software: An Update. *SIGKDD Explorations*, *1(2)*, 10–18.

Halliday, M. A. K. (1961). Categories of the Theory of Grammar. *Word,* 17, 241–292.

Herbst, T. & Mittmann, B. (2008). Collocation in English dictionaries at the beginning of the twenty-first century. In U. Heid, S. Schierholz, W. Schweickard, H. E. Wiegand & W. Wolski (Eds.), *Lexicographica* (pp. 103–119). Tübingen: Niemeyer.

Jin, P., Sun, X., Wu, Y. & Yu, S. (2007). Word clustering for collocation-based word sense disambiguation. In P. Jin, X. Sun, Y. Wu & S. Yu (Eds.), *Lecture notes in computer science: Computational linguistics and intelligent text processing* (pp. 267–274). Berlin: Springer-Verlag.

Kilgarriff, A., Rychly, P., Smrz, P. & Tugwell, D. (2004). The Sketch Engine. In G. Williams & S. Vassier (Eds.), *Proceedings of EURALEX* (pp. 105–116). France: Université de Bretagne Sud.

Mel'čuk, I. (1974). *Opyt teorii lingvističeskix modelej "Smysl ↔ Tekst". 'A Theory of the Meaning-Text Type Linguistic Models'*. Moskva: Nauka.

Mel'čuk, I. (1995). Phrasemes in language and phraseology in linguistics. In M. Everaert, E.-J. van der Linden, A. Schenk & R. Schreuder (Eds.), *Idioms: Structural and Psychological perspectives* (pp. 167–232). Hillsdale, NJ: Lawrence Erlbaum.

Mel'čuk, I. (1996). Lexical functions: A tool for the description of lexical relations in a lexicon. In L. Wanner (Ed.*), Lexical functions in lexicography and natural language processing* (pp. 37–102). Amsterdam, Philadelphia: Johm Benjamins.

Porzig, W. (1934). Wesenhafte Bedeutungsbeziehungen. *Beträge zur Geschichte der deutsche Sprache und Literatur*, 58, 70-97.

Provost, J. (1999) *Naive-Bayes vs. Rule-Learning in classification of email. Technical report AI-TR-99-284*. Austin, Texas: University of Texas at Austin.

The University of Waikato. (2010a). The University of Waikato Computer Science Department Machine Learning Group, WEKA Manual for Version 3-6-2, [online]. Retrieved from: http://iweb.dl.sourceforge.net/roject/weka/documentation/3.6.x/WekaManual-3-6-2.pdf

The University of Waikato. (2010b). The University of Waikato Computer Science Department Machine Learning Group, WEKA download [online]. Retrieved from: http://www.cs.waikato.ac.nz/ ~ml/weka/index_downloading.html

Vossen P. (Ed). (1998). *EuroWordNet: A multilingual database with lexical semantic networks*. Dordrecht: Kluwer Academic Publishers. The Spanish WordNet [online]. Retrieved from: http://www.lsi.upc.edu/~nlp/web/index.php?Itemid=57&id=31&option= com_content&task=view and http://adimen.si.ehu.es/cgi-bin/wei/public/wei.consult.perl

Wanner, L. (2004). Towards automatic fine-grained semantic classification of verb-noun collocations. *Natural Language Engineering*, *10*(2), 95–143.

Weinreich, U. (1969). Problems in the analysis of idioms. In J. Puhvel (Ed.), *Substance and structure of language* (pp. 23–82). CA, Los Angeles: University of California Press.

Wehrli, E., Seretan, V., Nerima, L. & Russo, L. (2009). Collocations in a rule-based MT system: A case study evaluation of their translation adequacy. *Proceedings of the 13th Annual Meeting of the European Association for Machine Translation*. Barcelona: Spain.

Williams, G. (2002). In search of representativity in specialised corpora: Categorisation through collocation. *International Journal of Corpus Linguistics*, 7, 43–64.

Witten, I. H. & Frank, E. (2005). *Data mining: Practical machine learning tools and techniques*. San Francisco: Morgan Kaufmann.

WordNet Release 2.1. (2005). [online]. Retrieved from: http://wordnet.princeton.edu

Zhong, Z. & Tou Ng, H. (2010). It makes sense: A wide-coverage word sense disambiguation system for free text. *Proceedings of System Demonstrations, 48th Annual Meeting of the Association for Computational Linguistics.* Sweden, Uppsala: Uppsala University.

NOTES

[1] A hyperonym of a word A is a word B such that B is a kind of A. For example, 'flower' is a name for 'rose', 'daisy', 'tulip', 'orchid', 'so flower' is hyperonym to each of those words. In its turn, hyperonym of 'flower is plant', and the hyperonym of 'plant' is 'living thing', and hyperonym of 'living thing' is 'entity'. Thus hyperonyms of a single word form a chain (rose → flower → plant → living thing → entity), and all words connected by the relation 'kind-of', or hyperonymy, form a tree.

[2] The Spanish Web Corpus is accessible only through the Sketch Engine, information on the corpus can be found at http://trac.sketchengine.co.uk/wiki/ Corpora/ SpanishWebCorpus/

[3] A notion of token is used mainly in computational linguistics. A token is a string of symbols in text separated by white spaces or punctuation marks. A token is not a word, but every concrete usage of a word, number, or other symbol in text. For example, in the sentence 'I saw him but he did not see me' there are nine tokens but eight words ('saw' and 'see' is the same word used in different tense forms). Sometimes, speaking of a corpus, the term 'word' is used in the meaning of 'token', for example, 'This corpus contains a million of words'.

[4] This list is available at www.Gelbukh.com/lexical-functions/

[5] WordReference Online Dictionary is available at http://www.wordreference.com/