Automatic Acquisition of Synonyms of Verbs from an Explanatory Dictionary using Hyponym and Hyperonym Relations

Noé Alejandro Castro-Sánchez and Grigori Sidorov

Natural Language and Text Processing Laboratory,
Center for Research in Computer Science (CIC),
Instituto Politécnico Nacional (IPN),
Av. Juan Dios Batiz, s/n, Zacatenco, 07738,
Mexico City, Mexico
noe.acastro@gmail.com, sidorov@cic.ipn.mx

Abstract. In this paper we present an automatic method for extraction of synonyms of verbs from an explanatory dictionary based only on hyponym/hyperonym relations existing between the verbs defined and the genus used in their definitions. The set of pairs *verb-genus* can be considered as a directed graph, so we applied an algorithm to identify cycles in these kind of structures. We found that some cycles represent chains of synonyms. We obtain high precision and low recall.

Keywords: automatic acquisition of synonyms, hyponym and hyperonym relations, directed graph, cycles in explanatory dictionaries.

1 Introduction

Dictionaries are very important linguistic resources that contain the language vocabulary and allow its automatic processing.

There are various kinds of dictionaries and various ways to classify them. In this research we focus on dictionaries aimed at natives of a language (monolingual), without domain restrictions with the registered vocabulary (general) and that present the semantic definition of the lexical entries (explanatory).

Dictionaries present textual sections known as Lexicographic Article (LgA) that consists of an entry named *Lexical Unit* (LU) and the information that defines it or describes it. The information contains the elements that show the constraints and conditions for the use of the LU, and the semantic information (or definition) which represents the basic content of the LgA.

Very well known norms are followed for constructing definitions for the content words (what we primarily are interested in), which are named as *Aristotelic Definition*. It consists in a sentence headed by a generic term or hyperonym (*genus*) followed by characteristics that distinguish the LU from other items grouped within the same genus (*differentia*).

In this work we focus in this kind of lexical relations given between the LU (hyponym) and the genus (hyperonym) used in its definition. We considered all the pairs LU-genus as a directed graph, and then we applied an algorithm to find all the elementary cycles. We found that some of these cycles are made up for verbs that are synonyms.

This approach is similar to other recent works which consider dictionaries as graphs, linking headwords with words appearing in their definitions. In [2] a graph is constructed from a dictionary based on the assumption that synonyms use similar words in their definitions. The vertexes of the graph are words of the dictionary and an edge from vertex *a* to vertex *b* shows that word *b* appears in the definition of *a*. In [7] the graph structure of a dictionary is considered as a Markov chain whose states are the graph nodes and whose transitions are its edges, valuated with probabilities. Then the distance between words is used to isolate candidate synonyms for a given word. The work [5] uses multiple resources to extract synonymous English words, like a monolingual dictionary, a parallel bilingual corpus (English-Chinese) and a monolingual corpus. Each resource was processed with a different method to extract synonyms and then an ensemble method was developed to combine the individual extractors. In [11] it is argued that definitions in dictionaries provide a regular syntax and style information (definitions) which provide a better environment to extract synonyms. It is proposed three different methods, two rule-based ones using the original definitions texts and one using the maximum entropy based on POS-tagged definitions.

The paper is organized as follows. In section 2, we explain how we process the dictionary and how we process the genus in the different ways they are used. In section 3, the method of creation of the graph is presented. In section 4, we show the results of our method, explain how we got the synonyms from a dictionary of synonyms for comparison and discuss the results. Finally in section 5, we conclude our studies and propose directions of the future work.

2 Processing of Dictionary

For our experiments the dictionary of Spanish Royal Academy (DRAE, as is known in Spanish) is used. It contains 162,362 definitions (senses) grouped in 89,799 lexical entries. From these, 12,008 lexical entries correspond to verbs, which contain 27,668 definitions (senses).

In this work, we are processing only verbs. We extract them from the dictionary, and then tagged them with the FreeLing parser, an open source text analysis tool for various languages including Spanish [1].

The next step was to identify and separate the grammatical marks, notes on usage, and other elements in the LgA.

2.1 Extraction of Genus from Definitions

Almost all definitions included in the dictionary follow the typical formula represented by *genus* + *differentia* (see Section 1). The predictable position of these elements allowed us to identify them in an automatic way.

Genus can be found in different ways, as it is shown below (in some cases the language differences between English and Spanish do not allow showing the characteristics in question):

1. As an only verb:
   Cotizar: *Pagar* una cuota.
   (Pay: *Pay a* cuote.)
2. As a chain of verbs linked by conjunctions or disjunctions:
   Armonizar. *Escoger y escribir* los acordes correspondientes a una melodía. (**Harmonize**. *Choose and write* chords for a melody).
   Aballar. *Amortiguar, desvanecer o esfumar* las líneas y colores de una pintura.
   (Disappear. *Disappear or vanish* the lines or colors of a paint*)*
3. As a subordinate clause in infinitive carrying out the function of direct complement.
   Gallear. *Pretender sobresalir* entre otros con presunción o jactancia.
   (**Brag**. *Pretend to excel* boastfully).
4. As a verbal periphrasis:
   Pervivir. *Seguir viviendo* a pesar del tiempo o de las dificultades.
   (**Survive**. *To remain* alive despite the time or difficulties).
5. As a combination of the previous points.
   Restaurar. *Reparar, renovar o volver a poner* algo en el estado que antes tenía.
   (**Restore**. To *repair, renovate or bring* back something to a previous state).

The items are shown in ascending order of complexity of processing. The items 1 and 2 are trivial. In 1 we identify the only verb and consider it as genus. In 2 we select all verbs that are heads of the clause as different genus. In items 3 and 4, we consider that the clause had only one genus made up of two verbs. Finally, in 5 we apply the previous considerations to identify the genus.

3 Construction of the Graph

We know that the relation between a LU and it genus is a hyponym-hyperonym relation. So, if we list all the pairs between LU-genus we obtain a directed graph, as is shown in the figure 1.

Each square represents a different verb and each number in circles is a different sense of a verb. So: *S* is a verb with senses *1* and *2*. *G1* is the genus (verb) of the sense 1's definition of verb *S*; *G2* is the genus for definition in sense *2* of *S*, and so on.

But, if each verb has different number of senses, we start from a specific sense of the hyponym verb, but we do not know to which sense of the hyperonym we should establish the relation. As there is no explicit information for solving this problem, we assume that the relation can probably be to the first sense of the hyperonym, because dictionaries present the most common used sense in the first sense (see Section 4 for another possibility).

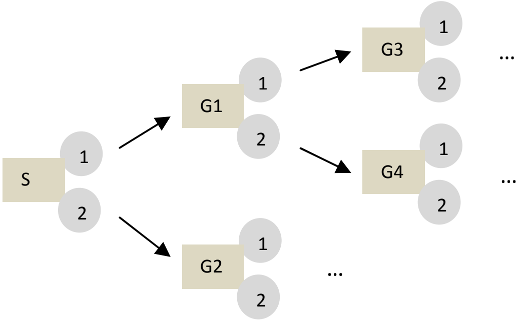

Fig. 1. Graph constructed from hyperonym relations

Now we can formalize these relations as:

$$V_{i=1}^{n} \rightarrow G_{j=1}(i)$$

Where:
V: Any verb.
i = Number of sense in *V* that is processed.
n = Total number of senses in *V*.
G = Genus of sense *i* in *V*.
j = First sense of Genus.

All this means that each sense of *V*, from *i* = 1 to *n*, is mapped to the first sense of Genus of the processed sense of the verb.

3.1 Extraction of Cycles

Obviously, any dictionary that defines all words it mentions must contain cycles (paths in which the first and the last vertices are identical); thus, cycles are an inevitable feature of a human-oriented dictionary that tries to define all words existing in the given language [4]. But it is assumed that a graph created from hyponym-hyperonym relations cannot contain cycles. However while processing some of the verbs, it is possible to find quite the opposite. For example:

1. **Pasar**. (1) *Llevar, conducir* de un lugar a otro.
   (**Pass**. (1) To take, to convey from one place to other).
2. **Llevar**. (1) *Conducir* algo desde un lugar a otro…
   (**Take**. (1) *Convey* something from one place to other…).
3. **Conducir**. (1) *Llevar, transportar* de una parte a otra.
   (**Convey**. (1) *Take, transport* from one place to other).

4. **Transportar**. (1) *Llevar* a alguien o algo de un lugar a otro.
(**Transport**. (1). *Take* someone or something from one place to other).

Creating the graph, we obtained:

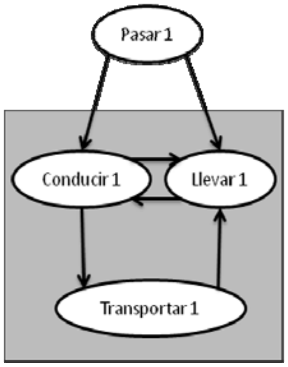

Fig. 2. Graph showing cycles among verbs linked from the genus of their definitions.

So, connection between *Conducir* and *Llevar* allows start of the path in some of them and finish in the same starting vertex. There is a longer cycle (understanding *length* as the number of vertices covered to reach the starting vertex), which include the vertexes *Conducir*, *Llevar* and *Transportar*.

If definitions of those verbs are analyzed, the cycle suggests a different semantic relation than hyponym/hyperonym, which is the relation of being a *synonym*. So, what we think is that some (aristotelic) definitions, at least in this dictionary, do not use a genus or hyperonym, but a synonym.

For identification of the cycles, for each verb in the dictionary: it was identified the genus in the first sense, and we created a path to the first sense of the genus. After repeating this process we identified some cycles that correspond to synonymy.

4 Evaluation

The process of obtaining synonyms from the hyponym/hyperonym relations produced the identification of 225 verbs grouped in 84 cycles. This means that exist 84 groups of synonyms. To measure precision and recall we used the Spanish Espasa's Dictionary of Synonyms and Antonyms (2005), which contains more than 200,000 synonyms and antonyms separated for senses and grammatical categories.

The precision of our method was of 0.92. The errors are related with the following: 0.03 of verbs were not found in Espasa's dictionary and 0.05 of verbs that were reviewed by hand represent real synonyms. For example, definitions given by DRAE of verbs "*Sumir*" and "*Hundir*" are:

1. **Sumir**. *Hundir* o meter debajo de la tierra o del agua.
(**Plunge**. To sink or put under the ground or water).

2. **Hundir**. *Sumir*, *meter* en lo hondo.
   (**Sink**. Plunge, put at depth).

In Espasa's dictionary, the only verbs having *sumir* as synonym are *abismar* and *sepultar*, although DRAE's definitions of both verbs show them as synonyms.

On the other hand, most of the cycles are made up for only two verbs, which gives a recall of 0.17 that is rather low. It is necessary to say that Espasa's dictionary does not provide an exhaustive review of the synonyms that represent each sense, i. e. one sense includes various synonyms that in a explanatory dictionary are separated in different senses. For example, for the verb *Poner*, DRAE contains:

1. **Poner**. Colocar en un sitio o lugar a alguien o algo.
   (**Put**. To place in a specified position someone or something).

In the synonyms dictionary we found as synonyms of *Poner* verbs like *enchufar* (*plug in*), *adaptar* (*adapt*), *instalar* (*install*), and so on. All of these verbs are related with *Poner* but in a sense that is not the main. We do not know yet how the percentage of this kind of situations affects the recall.

4.1 Selection of the Correct Synonyms

Espasa's Dictionary groups synonyms by senses, so the question is how we can know that we are comparing our group of synonyms with the right synonyms took from the Espasa's Dictionary.

Let us consider the following: the Dictionary was converted into a Database where the synonyms are grouped into two fields: *Headword* (Hw) that is any word and *Synonyms* (Syn) that contains the synonyms of Hw. This relation is not commutative in the dictionary. This is to say, if the word *A* is in Hw and the word *B* is in Syn, it is not guaranteed that exists the interchanged relation (*B* in Hw and *A* in Syn). So, we do the following:

After naming each of our suggested synonyms as *candidates*, we apply the next steps to each candidate in the Espasa's Dictionary:

1. Extract synonyms for candidate *c* (candidate in Hw1).
2. Extract the verbs having the candidate *c* as synonym (candidate in Hw2).
3. Intersect results of step 1 with results of step 2.
4. The group of synonyms (sense) that has a higher number of verbs gotten in step 3 represents the synonyms which we consider to compare with.

4.2 Possible Improvements of the Algorithm

The previous method (see 3.1) only allows finding of a relatively little number of synonyms, and does not guarantee the extraction of all of them.

Here we explain an idea of a future method that works on the different data (all word senses of the genus, as compared to the current implementation of the method that uses only the first word sense of the genus) and in this way can increase the recall.

For example, the next sequence can't be discovered:

1. **Manifestar**. (2) *Descubrir*, poner a la vista.
   (**Manifest**. *To uncover*, to bring to light).
2. **Descubrir**. (1) *Manifestar*, hacer patente.
   (**Uncover**. *To manifest*, to make evident).

Manifestar in sense 2 and *Descubrir* in sense 1, it cannot be found with the previous algorithm (see section 3).

So, the solution is mapping the verb to all the senses of it genus. It can be formalized in the following expression:

$$V_{i=1}^{n} \rightarrow G_{j=1}^{m}(i)$$

Where:
V: Any verb.
i = Number of sense in *V* that is processed.
n = Total number of senses in *V*.
G = Genus of sense *i* in *V*.
j = Each sense of the Genus.
m = Total number of senses in G.

Then, verb *V*, from *i* = 1 to *n*, is mapped to all Genus' senses of *i*.

To get this task, we used the Johnson's algorithm [6], which report a faster processing than the well-known Tarjan's [8], [9] and Tiernan's algorithms [10].

We did some adaptations to the algorithm for our data processing: the inputs are files that are created from a specific verb. Each line of the file is made up from the mapping between verbs in a specific sense to their genus in all senses. For example, let's say that we want to create the file from the verb *Manifestar* in its sense 1, that is:

Manifestar. (1) *Declarar*, dar a conocer.
(**Manifest.** (1) *To declare*, to make known formally).

So, the content of the file is the next:

```
manifestar'1|declarar'1
manifestar'1|declarar'2
manifestar'1|declarar'3
manifestar'1|declarar'4
manifestar'1|declarar'5
manifestar'1|declarar'6
manifestar'1|declarar'7
manifestar'1|declarar'8
manifestar'1|declarar'9…
```

For each file given as input, the algorithm creates another file containing the cycles.

The main problem with this approach is that some cycles generated by the algorithm do not contain correct synonyms. Let us see some lines of the output for the verb *manifestar*:

```
manifestar'2,poner'14,representar'3,manifestar'2,
manifestar'2,poner'17,hacer'25,representar'3,manifestar'2,
manifestar'2,poner'17,hacer'26,representar'3,manifestar'2,
manifestar'2,poner'17,hacer'41,representar'3,manifestar'2,
manifestar'2,poner'43,hacer'25,representar'3,manifestar'2, ...
```

Consulting the definitions of the verb/sense appearing in the first line of the list, we have:

1. **Manifestar**. (2) Descubrir, *poner* a la vista.
   (**Manifest**. Uncover, bring to light).
2. **Poner**. (14) *Representar* una obra de teatro o proyectar una película en el cine o en la televisión.
   (**Put**. (14) Perform a play or show a movie in the cinema or in the television)
3. **Representar**. (3) *Manifestar* el afecto del que una persona está poseída.
   (**Represent**. (3) Manifest the affect that a person has).

It is clear that the senses of the three verbs do not represent the same semantic situation, and the verbs are not synonyms (still, they can be synonyms in other senses).

But even with this kind of troubles, it is possible to see that in the verbs constituting the cycles there are more synonyms than we obtained with the first algorithm. For example, for the verb *Manifestar*, all the verbs that make up the cycles are shown below:

Manifestar (manifest)
Declarar (declare)
Hacer (make)
Ejecutar (execute)
Poner (put)
Representar (represent)
comunicar (communicate)
descubrir (discover)
exponer (expose)
presentar (present)
disponer (arrange)
mandar (order)
tener (have)
colocar (put)
contar (tell)
arriesgar (risk)

The synonyms of the verb *Manifestar* are shown in boldface. With this method it is possible to get more synonyms and improve the recall. Still, we should verify that the precision will not reduce.

5 Conclusions and Future Work

In this work we propose a method for identifying the synonyms of verbs using an explanatory dictionary. The method is based on hyponym-hyperonym relations between the verbs (headwords) and the genus used in their definitions. This approach allowed us to identify that some aristotelic definitions of verbs do not use a genus or hyperonym, but a synonym. Otherwise we cannot explain why a sequence of verbs constructed from hyperonym relations finish in the starting verb.

The method presents two variants: the former is based on the fact that the first sense defining a headword is the most commonly used, so we think that cycles constructed among the first senses of verbs guarantees that the verbs are synonyms (we did not identify an opposite case at least in the dictionary we use). On the other hand, it has the problem of a low recall. We programmed and evaluated this variant.

We also propose an idea of the second variant thinking in identifying groups of synonyms that cannot be detected using the first method. Our idea is that it will improve the recall. The manual analysis of the cycles obtained using this variant shows promising results, still its exact evaluation is future work. The question is to identify those cycles that the algorithm produces and that are not correct. Some of them include verbs used as *Lexical Functions* (LF), defined as functions that associate a word with a corresponding word such that the latter expresses a given abstract meaning indicated by the name of lexical function. Some method could be used to identify LF (for example [3]) and discard cycles that contain them.

The proposed methods have various lexicographic applications, for example, improvement of definitions of some verbs comparing them with those used in their synonyms, searching a difference between a real hyperonym in a group of synonyms, etc.

Acknowledgements. Work done under partial support of Mexican Government (CONACYT projects 50206-H and 83270, SNI) and National Polytechnic Institute, Mexico (projects SIP 20080787, 20091587, 20090772, 20100773, 20100668; 20111146, 20113295, COFAA, PIFI), Mexico City Government (ICYT-DF project PICCO10-120), and European Commission (project 269180).

References

1. Atserias, J., Casas B., Comelles, E., Gonzáles, M., Padró, L., and Padró, M. FreeLing 1.3: Syntactic and Semantic Services in an Open-Source NLP Library. In: Fifth international conference on Language Resources and Evaluation, Genoa, Italy (2006) http://www.lsi.upc.edu/ nlp/freeling.
2. Blondel, V., Senellart, P. Automatic extraction of synonyms in a dictionary, in Proceedings of the SIAM Text Mining Workshop, Arlington, VA, 2002.
3. Gelbukh, A., Kolesnikova, O. Supervised Learning for Semantic Classification of Spanish Collocations. Advances in Pattern Recognition (2010) 6256: pp. 362-371.
4. Gelbukh, A., Sidorov, G. Automatic Selection of Defining Vocabulary in an Explanatory Dictionary. Lecture Notes in Computer Science N 2276, 2002, ISSN 0302-9743, Springer-Verlag, pp. 300–303.

5. Hang, W., Ming Z. Optimizing synonym extraction using monolingual and bilingual resources. In Proc. International Workshop on Paraphrasing, (2003).
6. Johnson, D. Finding all the Elementary Circuits of a Directed Graph. SIAM Journal on Computing, Vol. 4, No. 1. (1975), pp. 77-84.
7. Muller, P., Hathout, N., Gaume, B. Synonym Extraction Using a Semantic Distance on a Dictionary. Proceedings of TextGraphs: The Second Workshop on Graph Based Methods for Natural Language Processing, (2006), pp. 65-72.
8. Tarjan, R. Depth-first search and linear graph algorithms. SIAM Journal on Computing. (1972), pp. 146–160.
9. Tarjan, R. Enumeration of the elementary circuits of a directed graph. SIAM Journal on Computing, (1973), pp. 211-216.
10. Tiernan, C. An efficient algorithm for finding the simple cycles of a finite directed graph. Comm. ACM, 13 (1970), pp. 722-726.
11. Wang, T.: Extracting Synonyms from Dictionary Definitions. In Recent Advances in Natural Language Processing (2009).