Automatic Syntactic Analysis for Detection of Word Combinations*

Alexander Gelbukh[1,2], Grigori Sidorov[1], Sang-Yong Han[2+], and Erika Hernández-Rubio[1]

[1] Center for Computing Research, National Polytechnic Institute,
Av. Juan Dios Batiz s/n, Zacatenco 07738, Mexico City, Mexico
{gelbukh, sidorov}@cic.ipn.mx, www.gelbukh.com

[2] Department of Computer Science and Engineering, Chung-Ang University,
221 Huksuk-Dong, DongJak-Ku, Seoul, 156-756, Korea
hansy@cau.ac.kr

Abstract. The paper presents a method for automatic detection of "non-trivial" word combinations in the text. It is based on automatic syntactic analysis. The method shows better precision and recall than the baseline method (bigrams). It was tested on a text in Spanish. The method can be used for enrichment of very large dictionaries of word combinations.

1 Introduction

The concept of word combination is related to the possibility of different words to appear together in the text connected by a syntactic link. The task is not computationally trivial because syntactically connected words can be linearly far from each other, i.e., separated by other words.

There are different types of word combinations. Some word combinations are fixed, like idioms, e.g., *to kick the bucket* or lexical functions like *to pay attention* [14]. In case of idioms and lexical functions, the meaning of the whole cannot be deduced from the meaning of the constituent words. In idioms, usually all words loose theirs meanings. As far as lexical functions are concerned, only one word (in case of our example, *attention*) keeps its meaning, while the other word (*to pay*) expresses standard semantic relation between actants of the situation. Detailed description of lexical functions can be found, for example, in [14] or other works by Mel'čuk. Since the meaning of the combinations is not a sum of the meanings of the words, there are severe restrictions for compatibility in lexical functions. Namely, if we want to express the given meaning and the words that conserves its meaning is known, then usually the choice of the other word is predetermined.

[*] Work done under partial support of Mexican Government (CONACyT, SNI), IPN (CGPI, COFAA, PIFI), Korean Government (KIPA Professorship for Visiting Faculty Positions in Korea), and ITRI of Chung-Ang University. First author is currently on Sabbatical leave at Chung-Ang University. We thank Prof. I. A. Bolshakov for useful discussion.

[+] Corresponding author.

A. Gelbukh, G. Sidorov, S. Y. Han, E. Hernandez-Rubio. Automatic syntactic analysis for detection of word combinations. Computational Linguistics and Intelligent Text Processing (CICLing-2004), Lecture Notes in Computer Science, N 2945, Springer-Verlag, 2004, pp. 240–244.

In free word combinations, the meaning of a whole is obtained by summing the meanings of the constituent words. Still, always there are semantic constraints for compatibility even in free word combinations. For example, for the verb *to eat*, it is expected that its dependent (direct object) will be certain kind of food, etc. Thus, some words have a broader compatibility than the other, for example, *to see* can be combined with practically any physical object, while *to read* only with something that contains written material (some metaphoric usages are possible also), etc.

For denoting some "important" word combinations, a term *collocation* is used. There is no commonly accepted definition of collocation. In a strict sense, only idioms and lexical functions are collocations because they contain the information that cannot be deduced. Nevertheless, this contradicts to common practice [1, 2, 15], when frequent word combinations are also considered collocations.

In NLP tasks, a dominating approach for defining collocations is based on the mutual information of words. We can see that some pairs of words have high conditional probability, i.e., if we encounter one word in the text, then the probability to encounter the other word is relatively high; while the conditional probabilities of the majority of randomly chosen word pairs are very low. It is called mutual information. This is purely statistic point of view that ignores the semantic properties of word combinations. In many works, collocations (in the sense of mutual information) are detected automatically [4, 9, 11, 12, 16, 18]. In [16], some syntactic heuristics are used additionally for filtering the obtained collocations. Nevertheless, these methods ignore the overwhelming majority of word combinations including lexical functions and idioms that do not have sufficiently high frequency.

Are idioms, lexical functions, and free word combinations useful in natural language processing? The answer for idioms and lexical functions is obviously positive. Free word combinations (even without sufficient mutual information) are also useful for many NLP tasks; see, for example, [3, 8, 13, 17]. This idea is supported by manual development of the dictionaries of word combinations [2, 5, 6, 7, 15].

In this paper, we propose a method of automatic detection of word combinations of different types based on the automatic syntactic analysis (parsing).

The proposed method can be used for semiautomatic enrichment of the dictionaries of collocations and free word combinations. For example, one of the largest dictionaries of this type is CrossLexica [5, 7, 8] that contains about 750,000 word combinations for Russian language. CrossLexica was compiled manually during about 14 years. We hope that our method can facilitate substantially the compilation process.

In the rest of the paper, we first describe the method of automatic detection of word combinations, evaluate its performance, and finally draw some conclusions.

2 Experimental Setting

We conducted our experiment for a randomly selected Spanish text available from Internet (*Cervantes Digital Library*). In our experiments, we used probabilistic parser and CF-grammar with unification for Spanish language described in [10].

In the experiment, we apply the program that performs syntactic analysis, obtains word combinations with corresponding relation between words, filters them, and stores them in a database.

For filtering, we use both syntactic and morphological features. Say, we filter out relations with pronouns and articles according to morphological filters. In addition, we apply syntactic filters, according to which only word combinations that have the following syntactic relations are left: verb-subject, verb-object (direct or indirect), noun-modifier (adjective or other noun), verb-modifier (adverb). The other syntactic relations are filtered out. The name of relation is stored as well.

Some special cases are: (1) coordinative relation (for example, *to read newspaper and magazine* should give two word combinations *to read newspaper* and *to read magazine*), so, we split the relation; (2) relation with preposition. In case of prepositions, we took the dependent word of the preposition and marked its relation with the head (master) word of the preposition. This is justified by the fact that prepositions usually express grammar relations (say, in some languages these relations can be expressed by grammar cases), so they are not important for lexical links. On the other hand, the choice of a preposition is important linguistic information. Therefore, in this case we store all three members.

We used as a baseline a method of gathering the word combinations that takes all word pairs that are immediate neighbors (bigrams). We incorporated certain intelligence into the baseline method. Namely, after the modification, it ignores the articles and takes into account the prepositions. Let us present an example of our analysis.

Mamá compró una torta pequeñita y un pastel con una bailarina en zapatillas de punta. (*Mother bought a little bun and a cake with a dancer in ballet-shoes.*)

The following syntactic dependency tree corresponds to this sentence. The dependent words are below the headword with the horizontal shift equal to the horizontal shift of the headword plus 1, e.g., the verb in the line 1 has dependents in the lines 2, 14, 15; the conjunction in the line 2 has dependents in the line 3 and in the line 6; etc.

Note that the words are normalizes morphologically. We used Spanish morphological analyzer described in [10].

We mark with bold the syntactic categories that are used in our grammar. They have natural interpretation: ***V*** stands for verb, ***N*** – for noun, ***SG*** – for singular, etc. For marking the name of syntactic relation, {} are used. Note that the name of relation is stored with the dependent word, because the head can have several dependents. In parenthesis (), there are the word and its lemma along with their translation into English, e.g. (*compró: comprar / bought : to buy*).

```
1  V(SG,3PRS,MEAN) ( compró: comprar / bought : to buy)
2   CONJ_C {obj}  ( y: y / and : and)
3    N(SG,FEM) {coord_conj} ( torta: torta / bun : bun)
4     ADJ(SG,FEM) {mod}  ( pequeñita: pequeñito / little : little)
5      ART(SG,FEM) {det} ( una: un / a : a)
6    N(SG,MASC) {coord_conj}  ( pastel: pastel / cake : cake)
7     PR {prep}  ( con: con / with : with)
8      N(SG,FEM) {prep} ( bailarina: bailarina / dancer : dancer)
9       PR {prep}  ( con: con / with : with)
10       N(PL,FEM) {prep}  ( zapatillas: zapatilla / shoes : shoe)
11        PR {prep}  ( de: de / of : of)
12         N(SG,FEM) {prep} ( punta: punta / point : point)
13      ART(SG,FEM) {det} ( una: un / a : a)
14  N(SG,FEM) {subj} ( mamá: mamá / mother : mother)
15  $PERIOD  ( .: .,)
```

The following word combinations were found in this sentence. Note that the word combinations 4 and 7 are filtered out due to the morphological filters.

1. *comprar* (obj) *torta*{Sg} (*buy* (obj) *bun*{Sg})
2. *comprar* (obj) *pastel* {Sg} (*buy* (obj) *cake* {Sg})
3. *torta* (mod) *pequeñito* (*bun* (mod) *little*)
4. *torta* (det) *un* (*bun* (det) *a*)
5. *pastel* (mod) [*con*] *bailarina* {Sg} (*cake* (mod) [*with*] *dancer* {Sg})
6. *bailarina* (mod) [*con*] *zapatilla* {Pl} (*dancer* (mod) [*with*] *shoe* {Pl})
7. *bailarina* (det) *un* (*dancer* (det) *a*)
8. *zapatilla* (mod) [*de*] *punta* {Sg} (*shoe* (mod) [*with*] *point* {Sg} *//= ballet shoe*)
9. *comprar* (subj) *mamá* {Sg} (*buy* (subj) *mother* {Sg})

We also store the information about morphological form of the dependent word in some cases (number for nouns; gerund/infinitive/finite for verbs) since this information may affect the compatibility. Note that the words are normalized anyway: e.g., we store *shoe* {Pl} instead of *shoes*. This can be necessary for further calculation of statistics with the possibility to take into account or ignore these morphological characteristics.

3 Experimental Results

The parsed text contains 741 words in 60 sentences. Average length of a sentence is 12.4 words. Apart, we marked syntactic relations in these sentences manually.

For the baseline, the total number of words is 588 because among 741 words there are 153 articles and prepositions in the sentences.

The following results were obtained. The total number of correct manually marked word combinations is 208. From these, 148 word combinations were found by our method. At the same time, the baseline method found correctly 111 word combinations. On the other hand, our method found only 63 incorrect word combinations, while the baseline method marked as word combinations 1175 pairs (588*2 – 1 = 1175), from which 1064 are wrong pairs (1175 – 111 = 1064).

These numbers give us the following values of precision and recall. Let us remind that precision is the relation of the correctly found to totally found, while recall is the relation of the correctly found to the total number that should have been found. For our method, precision is 148 / (148+63) = 0.70 and recall is 148 / 208 = 0.71. For the baseline method, precision is 111 / 1175 = 0.09 and recall is 111 / 208 = 0.53. It can be seen that recall of our method is better and precision is much better than those parameters of the baseline method.

The results of our method can be improved by developing better grammar for the Spanish language than the grammar that we use now.

4 Conclusions

We presented a method of automatic detection of word combinations of certain types. The method is based on the results of syntactic analysis. Syntactic and morphological filters are used to avoid the trivial word combinations.

The method was tested for Spanish and shows better precision and recall than the baseline bigram method that takes all word pairs that are immediate neighbors. In our case, the baseline method was improved by ignoring the articles and processing prepositions. The proposed method can be used for semiautomatic enrichment of dictionaries of word combinations and allows for making it much easier and faster.

References

1. Baddorf, D. S. and M. W. Evens. Finding phrases rather than discovering collocations: Searching corpora for dictionary phrases. In: *Proc. of the 9th Midwest Artificial Intelligence and Cognitive Science Conference (MAICS'98)*, Dayton, USA, 1998.
2. Bank of English. Collins. http://titania.cobuild.collins.co.uk/boe_info.html
3. Basili, R., M. T. Pazienza, and P. Velardi. Semi-automatic extraction of linguistic information for syntactic disambiguation. *Applied Artificial Intelligence*, 7:339-64, 1993.
4. Biemann, C., S. Bordag, G. Heyer, U. Quasthoff, C. Wolff. Language-independent methods for compiling monolingual lexical data. In: A. Gelbukh (Ed.) *Computational Linguistics and Intelligent Text Processing*. Lecture Notes in Computer Science, N 2945, Springer-Verlag, 2004 (this volume).
5. Bolshakov, I. A. Multifunction thesaurus for Russian word processing. In: *Proceedings of 4th Conference on Applied Natural language Processing*, Stuttgart, 1994, p. 200-202.
6. Bolshakov, I. A. Getting One's First Million… Collocations. In: A. Gelbukh (Ed.) *Computational Linguistics and Intelligent Text Processing*. Lecture Notes in Computer Science, N 2945, Springer-Verlag, 2004 (this volume).
7. Bolshakov, I. A., A. Gelbukh. A Very Large Database of Collocations and Semantic Links. In: Mokrane et al. (Eds.) *Natural Language Processing and Information* Systems (*NLDB-2000*). Lecture Notes in Computer Science 1959, Springer, 2001, p. 103-114.
8. Bolshakov, I. A., A. Gelbukh. Word Combinations as an Important Part of Modern Electronic Dictionaries. *Procesamiento del Lenguaje Natural*, No. 29, 2002, p. 47-54.
9. Dagan, I., L. Lee, and F. Pereira. Similarity-based models of word cooccurrence probabilities. *Machine Learning*, 34(1), 1999.
10. Gelbukh, A., G. Sidorov, S. Galicia Haro, I. Bolshakov. Environment for Development of a Natural Language Syntactic Analyzer. *Acta Academia 2002*, Moldova, 2002, p. 206-213.
11. Kim, S., J. Yoon, and M. Song. Automatic extraction of collocations from Korean text. *Computers and the Humanities* 35 (3): 273-297, 2001, Kluwer Academic Publishers.
12. Kita, K., Y. Kato, T. Omoto, and Y. Yano. A comparative study of automatic extraction of collocations from corpora: Mutual information vs. cost criteria. *Journal of Natural Language Processing*, 1(1):21-33, 1994.
13. Koster, C.H.A. Head/Modifier Frames for Information Retrieval. In: A. Gelbukh (Ed.) *Computational Linguistics and Intelligent Text Processing*. Lecture Notes in Computer Science, N 2945, Springer-Verlag, 2004 (this volume).
14. Mel'čuk, I.Phrasemes in language and phraseology in linguistics. In: *Idioms: structural and psychological perspective*, pp. 167-232.
15. *Oxford collocation dictionary*, Oxford, 2003.
16. Smadja, F. Retrieving collocations from texts: Xtract. *Computational linguistics*, 19 (1):143-177, March 1993.
17. Strzalkowski, T. Evaluating natural language processing techniques in information retrieval. In: T. Strzalkowski (ed.) Natural language information retrieval. Kluwer, 1999.
18. Yu, J., Zh. Jin, and Zh. Wen. Automatic extraction of collocations. 2003.