A "not-so-shallow" parser for collocational analysis

Basili R.(*), M.T. Pazienza (*), P. Velardi (§)

(*) Dipartimento Ingegneria Elettronica,
Università di Roma,Tor Vergata
{rbas,pazienza}@tovvx1.ccd.utovrm.it

(§) Istituto di Informatica, Università di Ancona
vela@anvax2.cineca.it

Abstract. Collocational analysis is the basis of many studies on lexical acquisition. Collocations are extracted from corpora using more or less shallow processing techniques, that span from purely statistical methods to partial parsers. Our point is that, despite one of the objectives of collocational analysis is to acquire high-coverage lexical data at low human cost, this is often not the case. Human work is in fact required for the initial training of most statistically based methods. A more serious problem is that shallow processing techniques produce a noise that is not acceptable for a fully automated system.

We propose in this paper a not-so-shallow parsing strategy that reliably detects binary and ternary relations among words. We show that adding more syntactic knowledge to the recipe significantly improves the recall and precision of the detected collocations, regardless of any subsequent statistical computation, while still meeting the computational requirements of corpus parsers.

1. Week methods for the analysis of collocations

In the past few years there has been a flourishing of interest in the study of word collocations. A common method to extract collocations is using windowing techniques for the extraction of word associations. In (Zernik 1990; Calzolari and Bindi 1990; Smadja 1989; Church and Hanks 1990) associations are detected in a ±5 window. A wider window (±100 words) is used in (Gale et al. 1992). Windowing techniques are also used in (Jelinek et al, 1990), where it is proposed a trigram model to automatically derive, and refine, context-free rules of the grammar (Fujisaki et al, 1991).

Windowing techniques weekly model the locality of language as well as other lexical information. The reliability of the acquired information depends upon the window size. A small window fails to detect many important word relations, while enlarging the window affects the tractability of the statistical model (especially for markovian n-gram models). Finally, window-based collocations provide limited information when dealing with a variety of lexical phenomena. For example, the simple observation of word cooccurrences is not a suitable marker of lexical subcategorization.

Another popular approach is using a partial parser, augmented with statistical parameters. The reciprocal contribution of syntax and statistics has been outlined in (Zernik 1991) to have an important role for automatic lexical acquisition. The syntactic relations are usually derived by pre-processing the target corpus with a part-of-speech tagger or with a simplified parser. Syntactic markers are applied to elementary links among words or to more structured contexts. The partial character of the different parsers described in literature makes it possible to process large corpora at a "reasonable" computational effort.

Most syntax-based statistical approaches use deterministic parsing, derived from Marcus' work on PARSIFAL parser (Marcus, 1980). PARSIFAL is a deterministic parser with look-ahead capabilities, that enables partial analyses. One of the PARSIFAL emanations, the Fidditch parser by Hindle, is used in (Hindle 1990) to detect subject-verb-object (SVO) triples. SVO triples are allowed to be incomplete, i.e. the subject or the object can be missing. Noisy data (i.e. words that are neither syntactically nor semantically related) are reduced by the use of statistical measures, such as the *mutual information* (Church et al, 1991), as defined in information theory.

The Fidditch parser requires a lexicon including information about base word forms and syntactic constraints (e.g. the complement structure of verbs). Non-trivial preliminary work is thus necessary in tuning the lexicon for the different domains and sublanguages. A second problem with the Fidditch parser is poor performances: the recall and precision at detecting word collocations are declared to be as low as 50%: However it is unclear if this value applies only to SVO triples, and how it has been derived. The recall is low because the Fidditch parser, as other partial parsers (Sekine et al, 1992; Resnik and Hearst, 1993), only detect links between adjacent or near-adjacent words.

Though a 50% precision and recall might be reasonable for human assisted tasks, like in lexicography, supervised translation, etc., it is not "fair enough" if collocational analysis must serve a fully automated system. In fact, corpus linguistics became a popular research field because of the claim that shallow techniques could overcome the lexical coverage bottleneck of traditional NLP techniques. Among the applications of collocational analysis for lexical acquisition are: the derivation of syntactic disambiguation cues (Basili et al. 1991, 1993a; Hindle and Rooths 1991,1993; Sekine 1992) (Bogges et al. 1992), sense preference (Yarowski 1992), acquisition of selectional restrictions (Basili et al. 1992b, 1993b; Utsuro et al. 1993), lexical preference in generation (Smadjia 1991), word clustering (Pereira 1993; Hindle 1990; Basili et al. 1993c), etc.

In the majority of these papers, even though the (precedent or subsequent) statistical processing reduces the number of accidental associations, very large corpora (10,000,000 words) are necessary to obtain reliable data on a "large enough" number of words. In addition, most papers produce a performance evaluation of their methods but do not provide a measure of the coverage, i.e. the percentage of cases for which their method actually provides a (right or wrong) solution. It is quite common that results are discussed only for 10-20 cases.

In our previous papers, we used semantic tagging to further reduce the noise and gain evidence of recurrent phenomena even with small corpora. However, no accurate or shallow method can resume valid information that has been lost in previous steps (i.e. in extracting collocations). We believe that a higher precision and recall of the input collocational data is desirable to ensure a good coverage to the whatever lexical learning algorithm.

In this paper we describe a not-so-shallow, multi-step, parsing strategy that allows it to detect long distance syntactic relations while keeping the temporal complexity compatible with the computational requirements of large-scale parsers. We demonstrate that a bit more syntax can be added to the recipe, with a significant improvement over existing partial parsers. We do not discuss of any subsequent processing (statistically or/and knowledge based) that may be applied to further improve the quality of collocational data, since this is outside the scope of this presentation. The interested reader may refer to our previous works on the matter.

2. A "not-so-shallow" parsing technique

Our syntactic analyzer (hereafter SSA) extracts partial syntactic structures from corpora. The analyzer, based on discontinuous grammar (Dahl,1989), is able to detect binary and ternary syntactic relations among words, that we call *elementary syntactic links (esl)*, The framework of discontinuous grammars has several advantages: it allows a *simple notation,* and exhibits *portability* among different logic programming styles. The presence of *skip* rules makes it possible to detect long distance dependencies between co-occurring words. This is particularly important in many texts, for the presence of long coordinate constructions, nested clauses, lists, parenthesised clauses.

The partial parsing strategy described hereafter requires in input few more than a morphologic lexicon (section 2.1). Post morphologic processing, as described in section 2.2, is not strictly required, though obviously it increases the reliability of the detected word relations. The lexicon used is purely morphologic, unlike for the Fidditch parser, neither it requires training, like in n-gram based models. This means that the shallow analyzer is portable by minimum changes over different domains. This is not the case with the deterministic partial parsing used in similar works. Furthermore the grammar rules are easy to tune to different linguistic subdomains. The analyzer enables the detection of different types of syntactic links among words: noun-verb, verb-noun, noun-preposition-noun, etc. This information is richer than just SVO triples, in that phrase structures are partitioned in more granular units.

The parsing method has been implemented for different corpora, which exhibit very different linguistic styles: a corpus of commercial activities (CD), in telegraphic style, a legal domain (LD) on taxation norms and lows, and remote sensing (RSD) abstracts. The latter is in English, while the former two are in Italian. The English application is rather less developed (a smaller morphologic lexicon, no post-morphology, etc.), however it is useful here to demonstrate that the approach is language independent. In this paper we use many examples from the RSD.

2.1 Morphology

The morphologic analyzer (Marziali, 1992) derives from the work on a generative approach to the Italian morphology (Russo, 1987), first used in DANTE, a NLP system for analysis of short narrative texts in the financial domain (Antonacci et al. 1989). The analyzer includes over 7000 elementary lemmata (stems without affixes, e.g. *flex* is the elementary lemma for *de-*

flex, in-flex, re-flex) and has been experimented since now on economic, financial, commercial and legal domains. Elementary lemmata cover much more than 7000 words, since many words have an affix.

An entry in the lexicon is as follows:

```
lexicon(lemma, stem, ending_class,
      syntactic_feature)
```

where `lemma` is the elementary lemma (e.g. *ancora* for *ancor-aggio* (*anchor-age*)), `stem` is the lemma without ending (*ancor*), `ending_class` is one over about 60 types of inflections. For example, *ancora* belongs to the class *ec_cosa*, since it inflects like the word *cosa (thing)*.

The Italian morphologic lexicon and grammars are fully general. This means that the analyzer has a tendency to overgenerate. For example, the word *agente* (*agent*, in the sense of *dealer*), is interpreted as a noun and as the present participle of the verb *agire (to act)*, though this type of inflected form is never found in both Italian domains. This problem is less evident in English, that is less inflected. Overgeneration is a common problem with grammar based approaches to morphology, as opposed to part of speech (pos) taggers. On the other side, pos taggers need manual work for corpus training every since a new domain is to be analyzed.

To quantitatively evaluate the phenomenon of overgeneration, we considered a test set of 25 sentences in the LD, including about 800 words. Of these 800, there were 546 different nouns, adjectives and verbs (i.e. potentially ambiguous words). The analyzer provided 631 interpretations of the 546 words. There were 76 ambiguous words. The overall estimated ambiguity is 76/546=0,139, while the overgeneration ratio is better evaluated by:

$$O = [631 - (546-76)]/76=161/76=2,11$$

2.2. Post morphological processing

The purpose of this module is to analyse compound expressions and numbers, such as compound verbs, dates, numeric expressions, and superlatives. Ad-hoc context free grammar have been defined. Post morphological processing includes also simple (but generally valid) heuristic rules to reduce certain types of ambiguity. There are two group of such rules:

(i) Rules to disambiguate ambiguous noun-adjective (N/Agg) interpretations (e.g. *acid*)

(ii) Rules to disambiguate ambiguous verb-noun (V/N) interpretations (e.g. *study*)

One example of heuristics for N/Agg is:

If N/Agg is neither preceded nor followed by a noun, or N/Agg, before a verb is reached, Then it is a noun.

Ex:. *..and sulphuric acid was detected*

Though examples are in English, post morphology has not been developed for the English language at the time we are writing.

After post-morphologic analysis, the 546 nouns, verbs and adjectives produced only 562 interpretations. The new overgeneration ratio is then

$$O'=(562-(546-76))/76=92/76=1,2$$

The estimated efficacy of the post-morphology is 161/92=1,75, about 50% reduction of the initial ambiguity.

2.3. The parser

The SSA syntactic analysis is a rewriting procedure of a single sentence into a set of elementary syntactic links (esl). The SSA is based on a discontinuous grammar, described more formally in (Basili et al. 1992a). In this section we provide a qualitative description of the rules by which esl's are generated.

Examples of esl's generated by the parser are: N_V (the subject-verb relation), V_N (the direct object_verb relation), N_P_N (noun preposition noun), V_P_N (verb preposition noun), N_Adj (adjective noun), N_N (compound) etc. Overall, we identify over 20 different esl's. There is a discontinuous grammar rule for each esl. A description of a rule used to derive N_P_N links is in Figure 1. This description applies by straightforward modifications to any other esl type (though some esl rules include a concordance test).

As remarked at the beginning of this section, skip rules are the key to extract long distance syntactic relations and to approximate the behaviour of a full parser. The first predicate LOOK_RIGHT of Figure 1 skips over the string X until it finds a preposition (prep(w2)). The second LOOK_RIGHT skips over Y until it finds a noun (noun(w3)).

Given an initial string `NL_segment`, BACKTRACK force the system to analyse all the possible solutions of the predicate LOOK_RIGHT (i.e. one-step rigth skips) to derive all the N_P_N groups, headed by the first noun (i.e. w1). For example, given the string:

low concentrations of acetone and ethyl alchool in acqueous solutions

the following N_P_N are generated:

concentration of acetone, concentration of alchool, concentration in solution, acetone in *solution, alchool in solution,*

all of which are syntactically correct.

```
SSA_rule( NL_segment, N_P_N )

 BEGIN
        REPEAT
             IF NL_segment  is EMPTY   THEN
                  EXIT;
             ELSE
              BEGIN
               NL_segment= (w1  Rest)
               IF ( noun(w1) ) THEN
                 BEGIN
                   LOOK_RIGHT(X, w2, Rest, New_Rest); %Rest=(X w2 New_Rest)
                   IF ( TEST_ON(X) AND prep(w2) ) THEN
                      BEGIN
                       LOOK_RIGHT( Y, w2,  New_Rest,  _); %New_Rest = (Y w3 _)
                       IF ( TEST_ON(Y) AND noun(w3) ) THEN

                               ASSERT( esl( N_P_N, w1,  w2,  w3));

                       BACKTRACK;
                      END;
                   BACKTRACK;
                 END
              POP w1 FROM  NL_segment;
             END
 END.
```

Figure 1: A description of an N_P_N rule

An uncontrolled application of skip rules would however produce unacceptable noise. The TEST_ON() are *ad hoc* heuristic rules that avoid uncontrolled skips. For example, TEST_ON(X) in Figure 1 verifies that the string X does not include a verb. Hence, in the sentence:

> *... the atmospheric code compared*
> *favourably with results ...*

the N_P_N(code,with,results) is not generated. In general, there is one-two different heuristic rule for each esl rule. Heuristic rules are designed to take efficient decisions by exploiting purely syntactic constraints. Such constraints are simple and require a minimum computational effort (essentialy, unification among simple structures). In some case, a lower recall is tolerated to avoid overgeneration. For example, the second TEST_ON(Y) rule of Figure 1 verifies that no more than two prepositions are skipped in the string Y. This rule stems from the observation that words located more than three prepositions apart, are rarely semantically related, though a full syntactic parser would eventually detect a relation. Hence, in the NL segment:

1% accuracy on the night side of the Earth with stars down to visual magnitude tree

the triple (*accuracy,to,tree*) is not generated, though syntactically correct.

The derivation of esl's is enabled for non adjacent word by virtue of skip rules. However, interesting information can be lost in presence of more complex phenomena as nested relative clauses or coordination of phrase structures. To cope with these phenomena, a post syntactic processor has been developed to extract links stemming from coordination among previously detected links. This processing significantly increases the set of collected esl, and the quality of the derived lexical information. The contribution of this post syntactic processing device depends heavily on the structure of incoming sentences. In this phase, simple unification .mechanisms are used, rather than heuristics.

3. Performance evaluation

Recall and Precision

Many algorithms evaluate their recall and precision against a human reference performer. This pose many problems, like finding a "fair" test material, using a large number of judges to render the evaluation less subjective, and finally interpreting the results. One example of the latter problem is the following: in (Smadja 1993) the nature of a syntactic link between two associated words is detected *a posteriori*. The performance of the system, called XTRACT, we evaluated by letting human judges compare their choice against that of the system. The reported performances are about 80% precision, 90% recall. One such evaluation experiment is, in our view, questionable, since both the human judges and XTRACT make a decision outside the context of a sentence. The interpretation of the results then does not take into account how much XTRACT succeeds in identifying syntactic relations as they actually occurred in the test suite.

Another problem is that, a human judge may consider not correct a syntactic association on the ground of semantic knowledge[1]. Instead, the performance of a syntactic parser should be evaluated only on a syntactic ground.

We define the linguistic performance of SSA as its ability to approximate the generation of the full set of elementary syntactic links derivable by a complete grammar of the domain. Given the set Ω of all syntactically valid *esl* and the set ω of *esl* derived applying SSA, the *precision* of the system can be defined as the ratio

cardinality($\Omega \cap \omega$) / cardinality(Ω),

while its *recall* can be expressed by:

cardinality($\omega \cap \Omega$) / cardinality(ω),

Global evaluations of the *precision* and *recall* are estimated by the mean values over the whole corpora.

We designed for testing purposes a full attribute grammar of the Italian legal language, and we selected 150 sentences for which the full grammar was proved correct. For each parsed sentence, a program automatically computes the esl's globally identified (without repetitions) by the parse trees of each sentence, and compares them with those generated by SSA for the same sentence. The following Table gives a measure of performance:

Esl_type	RECALL	PRECISION
N_P_N	69.1 %	81.8 %
V_P_N	55 %	56 %
V_N	67.5 %	86.6 %
N_V	59 %	60.5 %

To fully appreciate these results, we must consider, first, that the evaluation is on a purely syntactic ground (many collocations detected by the full grammar and not detected by the SSA are in fact semantically wrong), second, that the domain is particularly complex. There is an average of 23 trees per sentences in the test set. In particular, the low performances of N_V groups (i.e. the subject relation) is influenced by the very frequent (almost 80%) presence of nested relatives (ex: The income that was perceived during 1988(..)is included..) and inversions (ex: si considerano esenti da tasse i redditi..=*it is considered tax-free the income..). No partial parser could cope with these entangled structures.

One interesting aspect is that these results seem very stable for the domain. In fact, incrementally adding new groups of sentences, the perfoemance values do not change significantly.

For completeness, we also evaluated the English grammar. In this case, the evaluation was carried entirely by hand, since no full grammar of English was available to automatically derive the complete set of esl's. First, a test set of 10 remote sensing abstracts (about 1400 words, 67 sentences) was selected at random. The results are the following:

Esl_type	RECALL	PRECISION
N_N	78 %	67 %
V_N	81 %	58 %
N_p_N	94 %	54 %
V_p_N	87 %	42 %
N_V	75 %	57 %

Here the recall is rather high, since sentences have a much simple structure. However, there are many valid long distance pp attachments that for example most existing partial parses would not detect. The precision is lower because the English parser does not have post morphology as yet. One major source of error at detecting N_V pairs are, as expected, compounds.

Complexity

The most important factors that influence the time complexity are: the number N of sentences (words) of the corpus and the number k of different discontinuous rules (about 20, as we said).

The global rewriting procedure of SSA depends on the length n of the incoming text segment according to the following expression:

$$\sum_{i=1}^{n} ke(n-i)$$

where e(x) is the cost of the application of a grammar rule, as for in Figure 1, to a segment of

[1] It is unclear whether Smadja considered this problem in his evaluation experiment

length x. e(x) is easily seen to depend on:

1. Predicates that test the syntactic category of a word (e.g. noun(wl)), whose cost is equal to that of a simple unification procedure i.e. τ;
2. TEST_ON predicates, whose cost is not greater than τ*n, where n is the substring length.

We can thus say that the expression e(x) of the complexity of SSA syntactic rules verifies the following inequality:

$$e(n) \leq 3\tau + 2\tau n = O(n)$$

Hence, the global cost is:

$$\sum_{i=1}^{n} ke(n-i) \leq \sum_{i=1}^{n} 3\tau k + 2\tau k(n-i) =$$

$$= 2\tau kn(n+1) + 3\tau kn = O(n^2)$$

A significant information is that the processing time needed on a Sun Sparc station by the full grammar to parse the test set of 150 sentences is 6 hours, while SSA takes only 10 minutes.

Portability and scalability

These two aspects are obviously related. The question is: How much, in terms of time and resources, is needed to switch to a different domain, or to update a given domain? Since we developed three entirely different applications, we can provide some reliable estimate of these parameters. The estimate of course is strongly dependent upon the specific system we implemented, however we will frame our evaluation in a way that broadly applies to any system that uses similar techniques.

Morphology:

Our experience when switching from the commercial to the legal domain was that, when running the analyzer over the new corpus, about 30,000 words could not be analyzed. This required the insertion of about 1,500 new elementary lemmata. Accounting for a new word requires entering the stem without affixes, the elementary lemma of the word and the ending class (see section 2.1). Entering a new word takes about 5-10 minutes when the linguist is provided with some on-line help, for example a list of ending classes, browsing and testing facilities, etc. With these facilities, updating the lexicon is a relatively easy job, that does not require a specialized linguist to be performed.

Clearly, when implementing several applications, the global updating effort tends to zero. This is not the case for statistically based part of speech taggers, that require always a fixed effort to train on a new corpus. On the long run, it seems that grammar based approaches to morphology have an advantage over pos taggers, in terms of portability.

Syntax

Our experience is that adding a new rule takes about one-two man days. First, one must detect the linguistic pattern that is not accounted for in the grammar, and verify whether it can be reasonably accounted for, given the intrinsic limitations of the parsing mechanism adopted. If the linguist decides that, indeed, adding a new rule is necessary and feasible, he/she implements the rule and test its effects. Grammar modifications are required to:

* Select the *esl* types of interests;
* Define the heuristic rules (TEST_ON), as discussed in Section 2.3.

One positive aspect of SSA is that its complexity is O(k) with respect to the number k of grammar rules. Hence adding new rules does not affect the complexity class of the method.

In summary, *portability* is an essential feature of SSA. While other parsers need a non trivial effort to be tuned on different linguistic domains, we need only minimal adjustment to ensure the required coverage of the morphologic lexicon. However, the activity of lexical extension is needed with every approach. Portability is also guarantied by the *modularity* of the approach.

4. Conclusions.

Shallow methods for corpus analysis claim to have several desirable features, such as limited manual work and high coverage. Our point is that this is not entirely true. Fully statistical methods require initial training over the corpus to estimate parameters, and this is not trivial. Most of all, the effort is exactly the same every since the domain changes. In addition, a lot of noisy data are collected unless some shallow level of linguistic analysis is added to increase performance. But even then, reliable data are collected only for a fragment of the corpus. And what about high coverage? On the other side, we wouldn't be here, had traditional NLP techniques had any chance to become truly scalable.

This paper showed, if not else, that a bit more syntax can be added to the recipe, while still meeting important requirements, such as computational complexity and portability. *In media stat virtus:* This could be the moral of this paper, and in general of our research on lexical acquisition. Of course, we don't know where exactly the perfect balance is, we just seek for a *better* balance.

References.

(Antonacci 1989), F. Antonacci, M.T. Pazienza, M. Russo, P. Velardi , (1989), A Logic based system for text analysis and lexical knowledge acquisition , in Data and Knowledge Engineering, vol 4.

(Basili et al. 1991), R. Basili, M. T. Pazienza, P. Velardi, (1991), Using word association for syntactic disambiguation, in Trends in Artificial Intelligence, E. Ardizzone et al., Eds., LNAI n. 549, Springer-Verlag.

(Basili et al. 1992 a) R. Basili, M. T. Pazienza, P. Velardi, (1992), *A shallow Syntax to extract word associations from corpora*", in Literary and Linguistic Computing, vol. 2.

(Basili et al. 1992 b) R. Basili, M. T. Pazienza, P. Velardi, (1992), *Computational Lexicons: the neat examples and the odd exemplars*, Proc. of 3rd. Conf. on Applied NLP.

(Basili et al.1993a), Basili, R., M.T. Pazienza, P. Velardi, (1993). Semi-automatic extraction of linguistic information for syntactic disambiguation, Applied Artificial Intelligence, vol. 4, 1993.

(Basili et al.1993b), Basili, R., M.T. Pazienza, P. Velardi, (1993). What can be learned from raw texts ?, Journal of Machine Translation, 8:147-173.

(Basili et al.1993c), Basili, R., M.T. Pazienza, P. Velardi, (1993). Hierarchical clustering of verbs, ACL-SIGLEX Workshop on Lexical Acquisition, Columbus Ohio, June.

(Bogges,1991), L. Bogges, R. Agarwal, R. Davis, Disambiguation of prepositional phrases in automatically labelled technical text (1991). Proc. of AAAI 1991

(Church and Hanks, 1990), K. Church and P. Hanks, Word association norm, mutual information and lexicography, Computational Linguistics, vol. 16, n.1, 1990

(Church et al, 1991), Church, Gale, Hanks and Hindle, Using statistics in lexical analysis, (1991). Lexical Acquisition, U. Zernik Ed., Lawrence Erlbaum Ass., Publ., 115-164.

(Calzolari and Bindi,1990) N.Calzolari and R. Bindi, Acquisition of lexical information from Corpora, (1990), Proc. of COLING 90.

(Dahl, 1989), Dahl,V., "Discontinous grammars", (1989). Computational Intelligence, n. 5, 161-179.

(Fujsaki et al.,1991) Fujisaki T., F. Jelinek, J. Cocke, E. Black, T. Nishino, A probabilistic parsing method for sentence disambiguation, (1991). Current trends in Parsing Technology, M. Tomita Ed., Kluwer Ac. Publ., 1991.

(Hindle and Rooths,1991) D. Hindle, M. Rooths, Structural Ambiguity and Lexical Relations (1991). Proc. of ACL 1991

(Hindle, 1990), D. Hindle, Noun Classification form predicate-argument structure (1990). Proc. of ACL 1990

(Hindle and Rooths,1991) D. Hindle, M. Rooths, Structural Ambiguity and Lexical Relations (1991). Proc. of ACL 1991

(Hindle and Rooths, 1993) D. Hindle and M. Rooths, Structural ambiguity and lexical relations (1993). Computational Linguistics, vol. 19, n. 1, 1993

(Gale et al, 1992), Estimating the upper and lower bounds on the performance of word-sense disambiguation programs, (1992). Proc. of ACL 1992

(Jelinek et al., 1990) F. Jelinek, J.D. Lafferty, R.L. Mecer, Basic methods of probabilistic context free grammars, (1990). Research Report RC16374 IBM YorkTown Heights NY 10598, 1990.

(Marcus, 1980), M. Marcus, A Theory of Syntactic recognition for Natural Language, MIT Press, 1980

(Marziali,1992), Marziali, A., "Robust Methods for parsing large-scale text archives, Dissertation, Facoltà di Ingegneria, Univerità "La Sapienza" Roma, a.a. 1992 .

(Pereira et al.'1993) F.Pereira, N. Tishby, L. Lee, (1993). "*Distributional Clustering of English Words*", in Proc. of ACL 93 Columbus, Ohio, June, 1993.

(Russo,1987), M. Russo, "A generative grammar approach for the morphologic and morphosyntactic analysis of the Italian language" (1987). 3rd. Conf. of the European Chapter of the ACL, Copenhaghen, April 1-3 1987.

(Sekine et al, 1992) Automatic learning for semantic collocations, (1992). Proc. of 3rd. ANLP, 1992

(Smadja,1989), F. Smadja, "Lexical cooccurences: the missing link", (1989). Literary and Linguistic Computing, vol.4, n.3, 1989.

(Smadja,1991), F. Smadja, From N-Grams to collocations: an evaluation of XTRACT, (1991). Proc. of ACL 1991

(Smadja,1990), F. Smadja, K. McKeon, Automatically extracting and respresenting collocations for language generation, (1990). Proc. of ACL 1990

(Smadja, 1993), F. Smadja, Retrieving collocations from text: XTRACT, (1993). Computational Linguistics, vol 19, n.1, 1993

(Resnik and Hearst, 1993) P. Resnik, M. Hearst, Structural Ambiguity and Conceptual Relations, (1993). Proc. of the workshop on Very Large Corpora, Columbus, June 1993

(Utsuro et al., 1993), T. Utsuro, Y. Matsumoto, M. Nagao, verbal case frame acquisition from bilingual corpora, (1993). Proc. of IJCAI 1993

(Yarowski, 1992) Yarowsky D., "*Word-Sense Disambiguation Using Statistical Models of Roget's Categories Trained on Large Corpora*", (1992). Proc. of COLING-92, Nantes, Aug. 23-28.

(Zernik,1990), U. Zernik, P. Jacobs, Tagging for Learning: Collecting Thematic relations from Corpus (1990). Proc. of COLING 1990

(Zernik,1991), U. Zernik, Ed. "Lexical Acquisition: Exploiting on-line resources to build a lexicon", (1991). Lawrence Erlbaum Publ., 1991.