Using Loglinear Clustering for Subcategorization Identification*

Nuno Miguel Marques (nmm@di.fct.unl.pt)**[1], Gabriel Pereira Lopes (gpl@di.fct.unl.pt)[1], and Carlos Agra Coelho (coelho@isa.utl.pt)[2]

[1] Dep. Informática - FCT/UNL
[2] Dep. Matemática - ISA/UTL

Abstract. In this paper we will describe a process for mining syntactical verbal subcategorization, i.e. the information about the kind of phrases or clauses a verb goes with. We will use a large text corpus having almost 10,000,000 tagged words as our resource material. Loglinear modeling is used to analyze and automatically identify the subcategorization dependencies. An unsupervised clustering algorithm is used to accurately determine verbal subcategorization frames. In this paper we just tackle verbal subcategorization of noun phrases and prepositional phrases. A sample of 81 Portuguese verbs was used for evaluation purposes 97% precision and 99% recall for noun phrases and 92% precision and 100% recall for prepositional phrases was obtained.

1 Introduction

Recent experiments led us to find that loglinear models can be used for clustering verbs and other words with similar subcategorization requirements [MLC98]. We will show how it is possible to extract subcategorization information from a tagged corpus by co-occurrence counting of certain part-of-speech tags in the corpus. Relative positional information of those tags will be taken into account. In this paper we will elaborate on verbal subcategorization but the same approach is also feasible for other syntactic categories. The only grammatical information supplied to our system was originated in a hand tagged corpus containing about 5000 words that was used to train a neural network tagger [ML96]. Then a larger corpus with almost 10,000,000 words was automatically tagged using this trained tagger. This larger tagged corpus was used for clustering purposes. It should be stressed that the used tags are word tags not phrase tags.

Other authors have also worked on subcategorization extraction. Michael Brent [Bre93] proposed an approach where each subcategorization frame could be extracted by using a small set of highly specific and discriminating cues (mainly pronouns and proper nouns). According to [Man93], these cues represented 3% of the interesting information for subcategorization information. More recently,

* Work supported by JNICT Projects CORPUS (PLUS/C/LIN/805/93) and DIXIT (2/2.1/TIT/1670/95)
** Work supported by PhD scholarship JNICT-PRAXIS XXI/BD/2909/94

Manning [Man93] and Briscoe and Carroll [BC97] instead of using Brent's cues used a part-of-speech tagger and a parser (a simple finite state parser by Manning and a wide coverage partial parser by Briscoe and Carroll) for counting phrases. The main problem with each of these approaches is the grammatical knowledge they require. Only previously known grammatical subcategorization patterns can be extracted and this can bias the analysis because verbs with unusual patterns will be systematically ignored.

Ushioda et all. [UEGW96] parses (using regular expression grammar rules) all sentences of a corpus containing a given verb. The frequency of use of a given rule after a verb was used to build a contingency table for that verb[Agr90]. By using a loglinear model for supervised statistical learning [Fra96], Ushioda et. all built a system that classifies the verbs according to the selection of the subcategorization frame. However supervision requires a corpus tagged with subcategorization information and even for English this is a problem since there is no annotated corpora carrying such information.

In this paper we show that unsupervised clustering, using loglinear models, can be applied to subcategorization extraction from automatically tagged corpora. Moreover, as we will discuss prior parsing of corpora is not mandatory. In the next section we will describe how loglinear independence models, [Agr90] can be applied to determine clusters of verbs subcategorizing the same type of phrase or clause. Then we will describe two distinct experiments that empirically evaluate the validity of the proposed methodology. Acquired clusters will be analysed and confronted with the information supplied by a Portuguese standard dictionary and by two subcategorization dictionaries. Finally conclusions will be drawn.

2 Independence Loglinear Model

Let's assume we have a set of counts for m features over any verb (v). In this paper we will use both the total number of verbal forms followed by a part-of-speech ($f(POS|v)$)[1] and the total number of verb forms in the corpus ($f(v)$). Based on these counts we can also determine the total number of verbs not followed by that part-of-speech ($f(\overline{POS}|v)$).

In the table bellow we present the frequencies of the pair article-noun (second column), article-absence of noun (third column) and absence of article (fourth column), for verbs *afirmar* (to assert) and *encontrar* (to meet).

	(art, n)	$(art, \overline{n})$	$\overline{art}$	$\widehat{\lambda^X}$
$v_{afirmar}$	514	379	7290	0
$v_{encontrar}$	413	320	6092	-0.1815
$\widehat{\lambda^Y}$	0	-0.2823	2.670	$\widehat{\lambda} = 6.225$

[1] As part-of-speech (POS) we will use article (art) or preposition a (to or at, denoted $prep(a)$).

This table is called a contingency table. Columns represent the feature counts and rows the verbs chosen for analysis. The statistical relations between the rows and columns in such a table can be analyzed by using loglinear models[Agr90]. The columns represent features counts: (art, n) counts the number of times that a given verb is immediately followed by the bigram article-noun; $(art, \overline{n})$ counts the number of times the verb is followed by an article and a part-of-speech different from noun $(f(v) - f(art, n|v))$; and $(\overline{art})$ the frequency of verbs not followed by article $(f(v) - f(art|v))$. The frequency for feature (art) (total frequency of articles after a given verb — $f(art|v)$) is calculated by adding the frequencies of features (art, n) and $(art, \overline{n})$.

The verbs (rows) in our table have an independent behavior regarding the chosen set of features. In this case the expected value for the observed counts in a contingency table could be estimated using the independence loglinear model [Agr90]:

$$logE_{ij} = \lambda + \lambda_i^X + \lambda_j^Y \qquad (i = 1, ..., I; j = 1, ..., J).$$

In this model $logE_{ij}$ is the logarithm of the expected frequency of cell (i, j) and equals the sum of a constant λ with a row parameter λ_i^X and a column parameter λ_j^Y. The estimated values of these parameters are represented respectively in the right column (headed by $\widehat{\lambda^X}$) and lower row (headed by $\widehat{\lambda^Y}$). The `GLIM` package (Numerical Algorithms Group 1986, [Hea88]) was used to fit the loglinear independence model to our data. When assuming independence, is easily shown [Agr90], that column parameters are related with the average of the column and that row parameters are related with the average of the row. The constant λ works as a scale parameter.

We can evaluate how good a model fits the available data by comparing the estimated values with the real ones. We will use the likelihood-ratio statistic:

$$G^2 = 2 \sum_{i=1}^{I} \sum_{j=1}^{J} O_{ij} \log(\frac{O_{ij}}{E_{ij}})$$

where O_{ij} is the observed frequency for cell (i, j). When a model holds, this statistic has a large-sample chi-squared distribution with $(I - 1)(J - 1)$ degrees of freedom. In the above example, $G^2 = 0.357322$, a value well bellow 5.991476 (the 95^{th} quartile of the chi-squared distribution with two degrees of freedom), i.e. we could not reject the independence assumption.

In [MLC98], we have shown how loglinear models can be used to find independent verb clusters. If we have a set of features $F_1, F_2, ..., F_r$, a cluster of verbs $\overrightarrow{v_1}$ and a candidate verb v_2, by modeling the contingency table $X =< F_1, F_2, ..., F_r >$, $Y =< \overrightarrow{v_1}, v_2 >$, we will be able to decide if verb v_2 has the same behavior regarding both the features $F_1, F_2, ..., F_r$ and the group of verbs $\overrightarrow{v_1}$. In [MLC98], we propose the following, very simple, Cobweb based clustering algorithm. This algorithm does not yet include Cobweb's merge and split operators [Fis87]:

1. Take a list of N verbs $V =< v_1, ..., v_N >$, occurring in a Corpus C, having for each verb v_i their frequency vector X_i (e.g. we could have $X_i =< freq(Art), freq(\overline{Art}) >_i$).
2. V is sorted by decreasing order on the sum of their features (e.g. $freq(V_i)$). The most informative verbs will be used to define our seed clusters.
3. set List-of-clusters to the most frequent verb.
4. For each v_i in V do
   (a) Join v_i to the group $\overrightarrow{v_j}$ in List-of-clusters where the independence model best explains the contingency table for Y, X (e.g. the table $Y =< \overrightarrow{v_j}, v_i >$, $X =< freq(Art), freq(\overline{Art}) >$ or $X =< freq(Prep), freq(\overline{Prep}) >$). We used the model's residual deviance p-value to measure the quality of the explanation: the verb will be added to the cluster where the achieved p-value is maximum.
   (b) If v_i doesn't fit with any models in $List - of - clusters$ add a new cluster containing v_i to the list of clusters.

3 Extracting Subcategorization Frames

The presence of a given phrase after a verb is usually signaled by the presence of certain syntactic constituents. For instances the presence of an article always signals the presence of a noun phrase, the presence of a preposition signals a prepositional phrase, a subordinated conjunction signals a subordinated clause. Infinitive form of verbs signals infinitive subordinated clauses. So, our basic assumption is: some part-of-speech tags are good clues for concluding about a subcategorization frame. Somehow we have taken the opposite approach to Brent[Bre93]. Instead of relying on highly accurate and specific cues (such as the pronoun *me*), we relay on very general and not less accurate clues (POS tags), as our experimental results will show.

In the remaining of this article we will evaluate our clustering algorithm ability for modeling subcategorization frames. Our focus will be on the description and discussion of these experiments.

3.1 Experimental Framework

A list of 3381 infinitive verb forms was automatically extracted from our 9,333,555 words tagged corpus. Every word tagged as a verb in the corpus was extracted and then reduced to its infinitive form by using the POLARIS [LMR94] lexicon (normally, a Portuguese verb has 60 distinct sinflected forms). For validation purposes we have assigned transitivity information to each verb in this list by using an electronic version of Porto Editora's dictionary. Two other dictionaries [VC92] and [Bus94] were also used to assign information about prepositional phrase subcategorization to some of the verbs in our list[2]. If we exclude transitivity information, these two dictionaries are, to the best of our knowledge, the only sources of subcategorization information for Portuguese. [VC92] covers

[2] The remaining verbs were assigned a subcategorization class by us, without special care regarding exhaustiveness.

1100 verbs and only informs about the prepositional subcategorization. [Bus94] presents the main subcategorization classes for 2000 verbs.

In the reported experiment for transitive verbs we have used the features $\overline{art}$ and art already described in previous section. In the Prepositional phrase experiments we have used the counts for Portuguese preposition a (*to* or *at*). This experiment will be denoted by features $prep(a)$ (counted by $f(prep(a)|v)$) and $(\overline{prep(a)})$ (counted by $f(v) - f(prep(a)|v)$). This preposition has two very interesting features: it is ambiguous between article, demonstrative pronoun, personal pronoun and preposition, so we are testing how does our approach support noise inserted by the part-of-speech tagger. Second it is one of the most frequent prepositions in Portuguese. So, we don't have to worry about scarce data.

3.2 The art, $\overline{art}$ Experiment

One of the most used verbal classifications distinguishes between transitive and intransitive verbs. It is assumed that a transitive verb subcategorizes a noun phrase. So, we have measured the frequency of articles appearing immediately after the verb (denoted by feature art). In order to know how frequent a verb is we have also measured the frequency of non articles occurring just after the considered verbs (denoted by feature $\overline{art}$). Table 1 synthesizes the acquired results after applying our algorithm to the selected list of verbs. In this table, second row, headed by *tr*, regards the verbs that are classified in clusters where the first element is a transitive verb. But we notice that there are 2 intransitive verbs classified as transitive. Row three refers to verbs that are classified as both transitive and intransitive in the consulted dictionaries. Verbs that were reported by rows 2 and 3 give rise to 22 clusters. Row 4 is related to verbs classified as intransitive in the consulted dictionaries. For these verbs we notice that 3 transitive verbs are clustered with intransitive verbs. Verbs that were both identified as transitive or intransitive, have been considered transitive just for our precision recall/evaluation[3]. Since the total number of transitive verbs in our sample was of 73 we have a 90% (73/81) global precision baseline over the dictionary and 88% over the corpus.

Inspecting the acquired clusters, we find that our reference dictionary is incomplete — verb *ser* (to be) is only classified there as intransitive. However this verb has a transitive nature in certain occurrences:

Este é **o terceiro dia do ano** *(this is* **the third day of the year***).*

Two transitive verbs are clustered with verb *ser*. The counts presented in table 1 have been corrected assuming that verb ser is in class *tr+intr*. The remaining three intransitive verbs clustered as transitive belong to the same cluster. This cluster has six verbs three transitive and three intransitive. It is

[3] As usual, precision is the percentage of correctly classified verbs (correctly classified verbs/total verbs) and recall is the percentage of classified verbs that were correct (correctly classified verbs/total of verbs classified).

	tr	tr+intr	intr	TtD	TtC	clusters	PrcD	PrcC
tr	33	4	2	74	225854	22	95%	97%
tr+intr	12	21	2					
intr	3	0	4	7	25631	2	57%	93%
TtD	73		8	81	—	24	—	—
TtC	220786		30699	—	251485	189076		
RclD	96%		50%	—			91%	—
RclC	99%		77%				—	97%

Table 1. Number of verbs in each type of cluster for the noun phrase experiment. Columns represent the dictionary data and rows the acquired clusters as evaluated by their first element. *C* stands for frequencies in the corpus, *D* for frequencies in the dictionary. *Tt* stands for total, *Prc* stands for precision and *Rcl* stands for recall. Columns and rows headed by *tr* represent transitive verbs, headed by *intr* intransitive verbs. *clusters* presents the total number of clusters.

headed by verb *vir* (to come, or to reveal). There is an explanation: Portuguese preposition *a* was wrongly tagged as An article, as in:

o caso veio **a** *público (the case was revealed* **to** *the public)*

Moreover, some forms of verb *vir* are identical to forms of verb *ver* (to see). Since verb *ver* is transitive, some articles are due to this yet unsolved lexical ambiguity. Another problem with some intransitive verbs is due to the exchange of positions between the verb and its subject - the verb appears before its subject:

veio **a velhice** *e chegou a vez dela* (she had grown old and her time has elapsed)

The remaining two intransitive verbs clustered as transitive were *caber* (to fit) and *funcionar* (to work, in the sense that something works). Most of the articles appearing conjointly with *caber* were due to wrong tagging of noun *cabo* as a verb in the Portuguese expression *levar a cabo* (to perform). In this expression noun *cabo* is usually followed by article (*levar a cabo a operação* — to perform the action). In some other cases preposition *a* was wrongly tagged as an article.

3.3 The *prep(a)*, $\overline{prep(a)}$ Experiment

Previous experiment was repeated for the same list of verbs using Portuguese preposition *a* to cluster our data. We used features *prep(a)* and $\overline{prep(a)}$. Results are shown in table 2. Again the second row, headed by *PP(a)*, regards the verbs that subcategorize phrases headed by preposition *a*. There are 16 verbs that don't subcategorize *PP(a)* but were incorrectly clustered as if they did. Row three regards verbs that don't subcategorize *PP(a)*. According to our data no errors were detected for these verbs. A 53% precision baseline over dictionary and corpus could be achieved by tagging all clusters as *dont* (the verb doesn't subcategorize prep(a)).

	PP(a)	dont	TtD	TtC	clusters	PrcD	PrcC
PP(a)	38	16	54	128323	17	70%	92%
dont	0	27	27	123162	4	100%	100%
TtD	38	43	81	—	21	—	—
TtC	118540	132945	—	251485	183794		
RclD	100%	63%	—			80%	—
RclC	100%	93%				—	96%

Table 2. Number of verbs in each type of cluster for the prepositional phrase experiment. *C* stands for frequencies in the corpus, *D* for frequencies in the dictionary. *Tt* stands for total while *Prc* stand for precision and *Rcl* stands for recall.

Just by looking at this table we found, that while identifying subcategorization in the presence of the preposition is fairly easy (there was no errors, and a 100% recall was achieved), identifying the absence of it is more difficult. Confirming this is the number of clusters needed to describe each pattern. We find much more distinct patterns in verbs with the preposition than in verbs without it. The algorithm needed 17 clusters in the first case and only 4 in the latter. These results conform with what could be expected: Verbs that don't subcategorize the preposition, co-occur less with it. This way, occurrences of the preposition are mainly due to chance, or to the presence of some complement.

There are 3 main causes of errors for clusters regarding verbs that subcategorize *PP(a)*: verb complements (mainly time and space locatives), tagger errors and low frequency errors. Tagger errors further subdivide into two types: verb tagging errors and argument tagging errors. Complements are a common cause of error. Some verbs just tend to co-occur too frequently with time complements. Example: verb *assinar* (to sign) occurs frequently with a date in our corpus, and is clustered as subcategorizing *PP(a)*.

As it was previously mentioned, Portuguese preposition *a* is ambiguous with the article *a*. In some cases the article (much more frequent than the preposition), is tagged as preposition *a*. This way, verbs subcategorizing a noun phrase, could be grouped in a *PP(a)* cluster. Fortunately, the tagger is extremely accurate in tagging prepositions, and so, few errors are due to this problem. The same does not occur with article *a*, example: verb *integrar* (to integrate), in the expression *integrar a força ...* (to integrate the [military] force), the article *a* is systematically tagged as a preposition. This error will probability be ameliorated in future versions of the tagger.

Incorrect identification of verbs is another cause for error. Nouns, tagged as verbs, could be counted as the verbal forms with which they are ambiguous. Example: town named *Caminha* was wrongly tagged as verb *caminhar* (to walk). Verb *caminhar* is not generally followed by preposition *a*, but name *Caminha* is usually followed by such preposition. This way, *caminhar* was wrongly grouped in a *PP(a)* cluster.

Low frequency errors refer to rare subcategorization frames of frequent verbs. So, occasional presence of the selected feature tends to cluster a less frequent non subcategorizing verb with a much more frequent subcategorizing one. Example:

in the cluster headed by *acrescentar* (to add), having seven verbs, the five less frequent ones have only one or two occurrences of feature (*prep(a)*). As a result these verbs have all been clustered as subcategorizing *PP(a)* verbs.

4 Conclusions

In related work, only Brent [Bre93], presents results specific for the subcategorization of noun phrases. A total of 66 verbs are identified having noun phrase arguments. Of these 63 were correct. Other 127 verbs had been manually identified as having a noun phrase argument. So this means a 49% recall and 95% precision. Brent used 5 subcategorization frames and obtained 96% precision and 76% recall. Other presented results in literature have smaller precision, but use a much richer subcategorization set. For instance Briscoe and Carroll ([BC97]) report 81% precision and 80% recall using more than two hundred subcategorization frames. Although comparisons are difficult (we are working in a different language, and we are evaluating our data by comparison with a dictionary, not manually, as Brent did), our acquired precision/recall results seem encouraging.

We think the algorithm that we have just presented is a good way to determine word subcategorization. The main drawback we have found was on low frequency verbs but this can be overcome by automatically looking for extra text having those verbs. Despite this we still expect to find some low frequency words due to Zipfs law. The best way to handle these verbs is probably by using a partial parser and model based fault finding but this is a complementary research problem. A small change in our algorithm may also be effective on low frequency verbs. First we should determine and evaluate the basic clusters for the most frequent verbs. Then a probability threshold P should be established, lets say at 95%. At that value the G^2 statistic could be used in hypothesis testing. A new verb should be considered tagged as belonging to all the clusters where the independence hypothesis couldn't reject it. We intend to evaluate this change to the algorithm for low frequency words soon.

We also intend to extend our algorithm in order to support a better search through our cluster space. For that we intend to insert cluster merging and cluster splitting operators, similarly to Fisher's Cobweb [Fis87]. Regarding the number of used features we are also presently researching for the effects of adding new dimensions to our contingency tables. One of the advantages of doing this by using loglinear models is the possibility of inserting interaction terms among the several features in our model. This way we will no longer need to assume statistical independence among our features[Fra96].

The best behavior of our algorithm was achieved when we counted for the presence of a certain unigram, bigram or trigram and its complement (that is the frequency of the verb minus the frequency of the feature) after the verb. We empirically found that the increase in the number of features tends to increase the used number of clusters. Similarly, if we don't use the complement of the features we have found that recall values were worst.

Additionally to the subcategorization frame, we have for each considered verb its expected value given by the loglinear model. By using this value we are providing frequencies that, although influenced by the verb subcategorization frame, are still particular to each verb. Our results, if we take into consideration verb relative frequencies in the corpus, are outstanding: 97% of all occurrences of transitive verbs are correctly identified, having a recall of 99%. In the prepositional phrase experience, 92% of precision was achieved without missing any verb that subcategorizes a prepositional phrase headed by the preposition under study. Moreover our approach has the additional advantage that almost no linguistic information is needed by our algorithm and so, it can be used as a tool for extracting subcategorization frames.

References

[Agr90] Alan Agresti. *Categorical Data Analysis.* John Wiley and Sons, 1990.

[BC97] Ted Briscoe and John Carroll. Automatic extraction of subcategorization from corpora. In *Proceedings of the 5th Conference on Applied Natural Language Processing (ANLP'97)*, 1997.

[Bre93] Michael R. Brent. From grammar to lexicon: Unsupervised learning of lexical syntax. *Computacional Linguistics*, 19(2):245–262, 1993.

[Bus94] Winfried Busse. *Dicionário Sintáctico de Verbos Portugueses.* Livraria Almedina, 1994.

[Fis87] D. H. Fisher. Knowledge acquisition via incremental conceptual clustering. *Machine Learning*, 2:139–172, 1987.

[Fra96] Alexander Franz. *Automatic Ambiguity Resolution in Natural Language Processing*, volume 1171 of *LNAI Series.* Springer, 1996.

[Hea88] M. J. R. Healy. *GLIM: An Introduction.* Clarendon Press, Oxford, 1988.

[LMR94] José Gabriel Lopes, Nuno Cavalheiro Marques, and Vitor Ramos Rocio. Polaris, a PORtuguese Lexicon Acquisition and Retrieval Interactive System. In *Proceedings of the conference on Pratical Applications of PROLOG*, 1994.

[Man93] Cristopher Manning. Automatic acquisition of a large subcategorization dictionary from corpora. In *Proceedings of the 31st Annual Meeting of ACL*, pages 235–242, 1993.

[ML96] Nuno C. Marques and José Gabriel Lopes. A neural network approach to part-of-speech tagging. In *Proceedings of the Second Workshop on Computational Processing of Written and Spoken Portuguese*, pages 1–9, Curitiba, Brazil, October 21-22 1996.

[MLC98] N.M.C. Marques, J.G.P. Lopes, and C. A. Coelho. Learning verbal transitivity using loglinear models. In *Lecture Notes in AI (LNAI): Proceedings of the 10th European Conference on Machine Learning.* Springer Verlag, Berlin, April 1998.

[UEGW96] A. Ushioda, D. Evans, T. Gibson, and A. Waibel. Estimation of verb subcategorization frame frequencies based on syntactic and multidimensional statistical analysis. In H. Bunt and M. Tomita, editors, *Recent Advances in Parsing Technology.* Kluwer Academic Publishers, 1996.

[VC92] Helena Ventura and Maunela Caseiro. *Dicionário Prático de Verbos Seguidos de Preposições.* Fim de Século Edições, LDA., 2 edition, 1992.