NLP for Shallow Question Answering of Legal Documents Using Graphs*

Alfredo Monroy[1], Hiram Calvo[1,2], and Alexander Gelbukh[1]

[1] Center for Computing Research, National Polytechnic Institute
Mexico City, 07738, Mexico
alopezm301@ipn.mx, hcalvo@cic.ipn.mx, gelbukh@gelbukh.com
[2] Nara Institute of Science and Technology, Takayama, Ikoma, Nara 630-0192, Japan
calvo@is.naist.jp

Abstract. Previous work has shown that modeling relationships between articles of a regulation as vertices of a graph network works twice as better than traditional information retrieval systems for returning articles relevant to the question. In this work we experiment by using natural language techniques such as lemmatizing and using manual and automatic thesauri for improving question based document retrieval. For the construction of the graph, we follow the approach of representing the set of all the articles as a graph; the question is split in two parts, and each of them is added as part of the graph. Then several paths are constructed from part A of the question to part B, so that the shortest path contains the relevant articles to the question. We evaluate our method comparing the answers given by a traditional information retrieval system—vector space model adjusted for article retrieval, instead of document retrieval—and the answers to 21 questions given manually by the general lawyer of the National Polytechnic Institute, based on 25 different regulations (academy regulation, scholarships regulation, postgraduate studies regulation, etc.); with the answer of our system based on the same set of regulations. We found that lemmatizing increases performance in around 10%, while the use of thesaurus has a low impact.

1 Introduction

Previous work [20] has shown that modelling relationships between articles of a regulation as vertices of a graph network works twice as better than traditional information retrieval systems for returning articles relevant to the question. Despite being that approach language independent, in this work we experiment by using natural language techniques such as lemmatizing and using manual and automatic thesauri for improving question based document retrieval. We focus in Spanish language. For automatic thesaurus we used a distributional thesaurus [19], and for the manual thesaurus we used a human oriented dictionary (Anaya) [21]. The advantage of using a distributional thesaurus is that the approach remains language independent—not being the

* We thank the support of Mexican Government (SNI, SIP-IPN, COFAA-IPN, and PIFI-IPN) and Japanese Government. Second author is a JSPS fellow.

A. Gelbukh (Ed.): CICLing 2009, LNCS 5449, pp. 498–508, 2009.

case with the human oriented dictionary. On the other hand, using a lemmatizer would make this approach language dependent, as for a particular a lemmatizer is needed. In particular, we want to measure the advantage of using these resources and we want to measure the possible benefit from adding this kind of information. Our system gives answers which consist of set of articles related to the question and also the relevant articles related with them to complement the answer. This is called shallow QA, because its operation lies in the middle of snippet retrieval and giving the exact answer.

We test our system with regard to a traditional vector space model information retrieval system to answer questions particularly for the Spanish language given a set of 25 regulation documents from the National Polytechnic Institute. For details of the rest of the System, see sections 2 and 3. The addition of NLP techniques is explained with further detail in Section 4. Evaluation and experiments are shown in Sections 5 and 6.

2 Related Work

There are not many works particularly devoted to the legal domain, despite of its wide use and application. Particularly for the legal domain, the workshop *Question Answering for interrogating legal documents* took place in 2003, in the framework of the JURIX Forum (The foundation for Legal Knowledge Based Systems). Several works showed that a common problem is that traditional Information Retrieval Methods are not adequate to find the relevant fragments which answer legal questions because they do not consider the logic relationships between articles. In addition, many questions require an answer which cannot be found explicitly in a single article, or fragments of them, but intrinsically in the relationship between articles [9,10]. Some works use logic inference mechanisms such as COGEX System [14] and the system by Quaresma *et al.* [7]. However, these systems need expensive resources such as ontologies, axioms, and are language dependent. To avoid such requirements, we use a graph for capturing the relationships between articles in regulations as proposed in [20].

3 System Design

The architecture of this SQAS is based on the work shown in [20]. It was designed considering common characteristics posed by regulation documents, as well as the kind of questions and answers expected by the user. It is important to mention that regulation texts have a defined structure, they are composed of chapters, and these, in turn, are subdivided in articles. This makes possible to use different techniques which with other kind of texts would not be possible. Articles from a single regulation text are related between them, and also there are links between different regulations.

We focus in questions where the answer can be given as a set of articles from a regulation. For example, for the question: *Is it possible to award a honourable mention to a bachelor if he chose to graduate using the qualification option?* the answer can be given as a set of articles: *See Chapter II of "On Graduating Options" and article 13, Chapter VII, "On the Professional exam", article 43.* When one looks to such articles, they say:

The option of graduating by qualification proceeds if the student's average is higher than 9.0 and all of his subjects were approved in an ordinary way.

The candidate can only aspire to the award of honorific mention, if, in addition to covering other requisites disposed in this regulation, he presents professional exam

From those articles, it can be concluded that Graduating by qualification does not require presenting the professional exam, so that it cannot be included within the article 43 fraction II of the mentioned Regulation; when this option is chosen, the candidate cannot obtain honorific mention.

This system does not return a completely logically evaluated answer such as 'yes' or 'no', which would need a more complex machinery; because of this, it is considered as part of the Shallow Question Answering Systems.

The fundamental parts of the system are shown in the graph in Figure 1. Question pre-processing consists on constructing the query based on the question, and Answer Extraction consists on adding the generated query as two new nodes (A and B). Then the shortest path between A and B is sought. We will show that this path contains articles highly related to the question, and they share certain degree of similarity between them. The following picture depicts this. A1, A2, ... A5 are articles, while A and B are parts of the question.

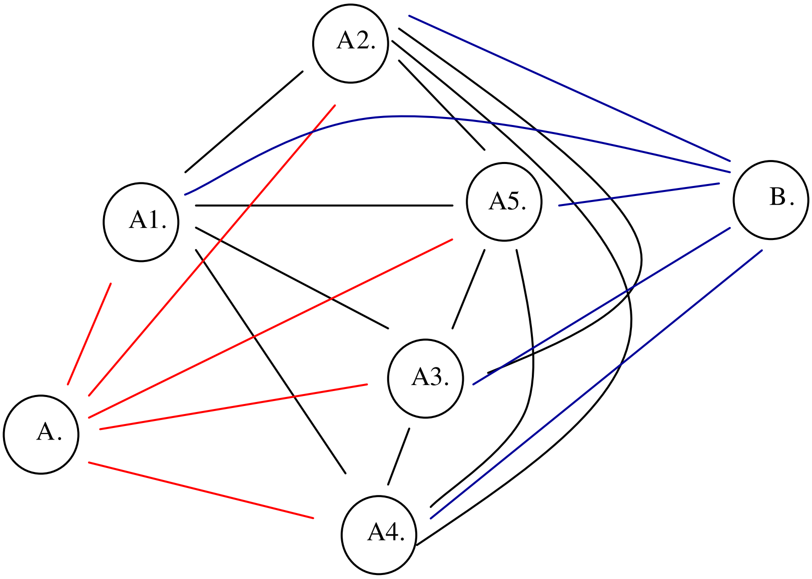

Fig. 1. Graph representing articles (A1, A2, ... A5) and question (A, B) for the Question Answering System

3.1 Graph Description

The documents (articles of regulations) were represented as vectors, following the Vector Space Model [12, 13]. Each term was weighted by the TF·IDF measure (Term Frequency Inverse Document Frequency). See equations (**1**), (**2**) and (**3**).

$$tfidf = tf_{t,j} \cdot idf_i \tag{1}$$

$$tf_{t,j} = \frac{n_{i,j}}{\sum n_{k,j}} \tag{2}$$

Where $n_{i,j}$ corresponds to the number of occurrences for each term from the article a_j and the denominator represents the occurrence of all terms in the article a_j

$$idf_i = \log \frac{|A|}{|\{a_j : t_i \in a_j\}|} \tag{3}$$

Where |A| is the total number of articles in the document collection and $|\{a_j : t_i \in a_j\}|$ is the number of articles where the term t_i appears.

Finally, a graph was constructed for each document collection, see Figure 2. Each node represents an article of a regulation text, and the associated values to the vertices $V_{i,j}$ between each pair of nodes represents the inverse value of the standard similarity cosine measure.

3.2 Question Processing

The question is pre-processed and then integrated into the graph in the same way that the regulation collection. Each question is converted to lowercase, punctuation symbols are eliminated (parenthesis, hyphens, numbers, etc.). Stopwords are kept (QK) or removed (QR) and finally, words that do not exist in the document collection are removed—this is equivalent to finding a similarity measure of 0 for them. The weighting values are calculated with regard to the document collection (See eq. (2)).

From the query of the previous module, two new *articles* are added to the graphs. The answer extraction consists on finding the paths of minimal weight using the Dijkstra algorithm: Once the first minimal path is found, the nodes which constitute it are eliminated, and the Dijkstra algorithm is run again until the graph becomes disconnected for the pair of nodes which constitute the query. The *answer paths* are ordered from less to more weight, and are returned to the user with the text of each article corresponding to the node in the regulation collection.

4 NLP Techniques for Shallow QA

Generally speaking, a thesaurus can be considered as a list of words with other words related to them. For example[1] *car* has the following **synonyms**: auto, automobile, machine, motor, motorcar, motor vehicle; and its **related words** are: bus, coach, minibus; beach buggy, brougham, compact, convertible, coupe, dune buggy, fastback, gas-guzzler, hardtop, hatchback, hot rod, jeep, limousine, roadster, sedan, sports car, station wagon, stock car, subcompact, van; flivver, jalopy.

[1] Example from the Merriam Webster on-line thesaurus (http://www.merriam-webster.com)/

Related words can be found generally in two ways:

- Manually built—Made by humans who collect the set of related words which they consider that can be associated to a specific word.
- Automatically built (distributional thesaurus)—Built automatically from a selected corpus. They obtain related words usually by comparing contexts on which the word is used [19]. For example, for the phrases *drink juice, drink lemonade, make juice, make lemonade, delicious juice, delicious lemonade*, then it might have enough evidence to conclude that *juice* and *lemonade* are related words.

On the other hand, thesaurus can be classified by their domain:

- Specific domain—Thesauri (manual or distributional) built based on a particular subject, *v. gr.* medicine.
- General domain—They have words from different areas, covering a general vocabulary.

Thesauri have been extensively used for query expansion [15, 16], so we expected that our system would benefit from their usage. We used mainly two tools: (1) A distributional thesaurus built from the Encarta Encyclopedia [17,18]; and (2) A manual thesaurus based on the Anaya dictionary [21].

We expand the query (with already omitted words): *go procedure representative election alumni council scholar* to, for example: *go transport take guide drive use procedure representative alumni student council meeting scholar educational.* Some words have a greater number of related words than others; some words are not present in the thesaurus, so they are not expanded.

We used two thesauri for these experiments:

1. A distributional thesaurus created from the Encyclopedia Encarta [17, 18]
2. A thesaurus based on the Anaya dictionary [21] using the synonyms from that dictionary.

Another NLP technique used in this work was lemmatizing. Although this is a language-dependent resource, many languages have a lemmatizer; moreover, it is possible to lemmatize unsupervisedly.

5 Evaluation

The evaluation of the QAS is based on the following criteria:

i Relevance: The output of the QAS should answer to the questions of the kind "yes or no"; this is measured by determining if the articles that the general lawyer considers to produce an answers were returned by the system.

ii Noise degree in the answer: Alien and irrelevant information that the system returns is quantified.

To implement the measure of these both criteria, we use the following procedure: Initially, the answer is limited to 100 articles. It is unlikely that the answer is found after such number of articles. The returned articles were divided in groups of 5 (the maximum number of articles that can be found in a single answer of the lawyer). Each group was given the value V_i, following equation (4)

$$V_i = 1 - \frac{(n-1)}{N} \tag{4}$$

Where n is the group number (the first 5 articles constitute the group 1, the next 5, group 2, etc). N is the maximum number of packages that can be found (N=20, as 5·20=100 articles).

Finally, each article returned for a determined answer is graded using the following expression:

$$Grade \quad CR_i = \frac{\sum_{i=1}^{n} nV_i}{n_{AR}} \tag{5}$$

Where CR_i is the grade assigned to the answer of the i-th question, n is the package number where the answer-article was found, and n_{AR} is the number of answer-articles found.

The final mark of the system is the average of the grades CR_i, i.e.:

$$Grade \quad CS = \frac{\sum_{i=1}^{np} CR_i}{np} \tag{6}$$

Where np is the total number of questions for evaluation.

6 Experiments and Results

For this report, our system was tested with 21 questions. We tested against a basic Information Retrieval System (IRS) based on the vector space model. This IRS uses the same set of vectors used for the construction of the graph, but this time they were directly compared with the cosine measure with the query. The vectors which are more similar to the query are returned, so that the output of this system is the set of articles relevant to the query. The results were then compared with the results of our QAS.

As we mentioned in Section 3.2, we tested with keeping and removing stopwords. This yields two derived regulation document collections: DCK (Document collection keeping stopwords) and DCR (Document collection removing stopwords), and two kinds of queries, which correspond to keeping or removing stopwords in the query.

Additionally, we performed four experiments with regard to the automatic division of the query in part A and part B. After the procedure described in sections 3.2, the query is divided in two new *articles* (nodes) A and B with the following contents, according to one of the four followings types of division:

Half Division (H): Node A will contain the left half of the question, and Node B will contain the rest. If the number of words in the query is odd, the Node A will contain one word more than Node B.
Mixed Division (M): In this type of division, terms are mixed: odd words are in Node A and even words are in Node B.

Reversed Half Division (H'): As in Half Division (H) but the contents of node A and B are exchanged.
Reversed Mixed Division (M'): As in Mixed Division (M) but contents of the Node A and Node B are exchanged. Even words are in Node A and odd words are in Node B.

Table 1. Results with no lemmatization

Stopwordst				Division	Precision	Answers found		Not found
Article List		Query						
Keep	Remove	Keep	Remove			Fully	Partially	
Our system								
LBA		✓		H	0.5342	12	5	4
				M	0.4988	13	3	5
				H´	**0.5351**	**12**	**5**	**4**
				M´	0.5023	13	3	5
			✓	H	0.4851	10	5	6
				M	0.4865	10	6	5
				H´	0.4858	9	6	6
				M´	0.4889	10	6	5
	LBA_R		✓	H	0.4603	9	6	6
				M	0.4723	10	5	6
				H´	0.4683	10	5	6
				M´	0.4716	10	5	6
IR system based on Vector Model Space								
LBA		✓		-	**0.2253**	**3**	**5**	**13**
	LBA_R		✓	-	0.1892	4	6	11

Table 2. Results using lemmatization

Stopwordst				Division	Precision	Answers found		Not found
Article List		Query						
Keep	Remove	Keep	Remove			Fully	Partially	
Our system								
LBA		✓		H	0.5170	11	8	2
				M	0.5536	12	8	1
				H´	0.5175	12	7	2
				M´	0.5548	12	8	1
			✓	H	0.5187	12	7	2
				M	0.6151	12	8	1
				H´	0.5175	12	7	2
				M´	0.5548	12	8	1
	LBA_R		✓	H	0.5026	13	6	2
				M	0.6103	13	7	1
				H´	0.5115	13	6	2
				M´	**0.6230**	**13**	**7**	**1**
IR system based on Vector Model Space								
LBA		✓		-	0.1976	2	5	14
	LBA_R		✓	-	**0.3055**	**5**	**7**	**9**

Table 3. Results using lemmatization and distributional Thesaurus

Stopwordst				Division	Precision	Answers found		Not found
Article List		Query						
Keep	Remove	Keep	Re-move			Fully	Partially	
Our system								
LBA		✓		H	0.5228	12	6	3
				M	0.4933	11	6	4
				H´	0.5253	12	6	3
				M´	0.4905	11	6	4
			✓	H	0.5250	12	6	3
				M	0.5138	12	5	4
				H´	**0.5273**	**12**	**6**	**3**
				M´	0.5126	12	5	4
	LBA_R		✓	H	0.5228	12	6	3
				M	0.4933	11	6	4
				H´	0.5253	12	6	3
				M´	0.4905	11	6	4
IR system based on Vector Model Space								
LBA		✓		-	0.1869	3	3	15
	LBA_R		✓	-	**0.2107**	**4**	**4**	**13**

Table 4. Results using lemmatization and the Anaya Thesaurus

Stopwordst				Division	Precision	Answers found		Not found
Article List		Query						
Keep	Remove	Keep	Re-move			Fully	Partially	
Our system								
LBA		✓		H	0.4689	12	6	3
				M	**0.5461**	**13**	**6**	**2**
				H´	0.4719	11	7	3
				M´	0.5456	13	6	2
			✓	H	0.4883	13	6	2
				M	0.5257	11	6	4
				H´	0.4856	12	7	2
				M´	0.5276	11	6	4
	LBA_R		✓	H	0.4736	12	7	2
				M	0.5325	12	6	3
				H´	0.4783	12	7	2
				M´	0.5336	12	6	3
IR system based on Vector Model Space								
LBA		✓		-	0.2428	4	4	13
	LBA_R		✓	-	**0.2815**	**4**	**7**	**10**

We tested every combination of these parameters. Table 1 shows the description and name of each experiment. We compared the performance of our system with two experiments based in the document collection DCK (Document Collection Keeping

Table 5. Summary of best results

Stopwordst				Division	Precision	Answers found		Not found
Article List		Query						
Keep	Remove	Keep	Remove			Fully	Partially	
With no lemmatization								
LBA		✓		H	0.5351	12	5	4
lemmatizing								
	LBA_R		✓	**DMT′**	**0.6230**	**13**	**7**	**1**
lemmatizing+ Anaya dictionary								
LBA		✓		DMT	0.5461	13	6	2
lemmatizing + distibutional thesaurus								
LBA			✓	DM′	0.5273	12	6	3

stopwords) and DCR (Document Collection Removing stopwords). These experiments are shown in the last two rows of Table 1.

7 Conclusions and Future Work

According to the reported values, the best result was found for the experiment where the list of articles and query keep non-content words, use division by the half, and walking from node B to node A. This is not significantly changed when walking from node A to node B.

Table 2 shows the results for the experiments where the article list was lemmatized. The best result was obtained for the experiment where the list of articles and query had stopwords removed. Precision is 0.6230 for this case. The precision when modifying the division type are very close, so that it has almost the same impact to traverse the nodes from A to B than doing so the reverse way. We compare with the traditional information retrieval system which obtains a precision of 0.3055. Lemmatizing the list of articles produced a rise in precision.

In Table 3, we show the results of adding a distributional thesaurus, in addition to lemmatizing the list of articles. The best result was found for the experiment where the article list and query keep stopwords. This value is 0.5461. The distributional thesaurus seems to have added noise, since the performance is inferior to previous experiments.

Finally, we experimented with using a manual thesaurus, which is based in the human oriented dictionary Anaya for Spanish (See Table 4). The best result was obtained when lemmatizing articles, keeping content words, but removing them from the query. The precision for this case was 0.5273.

From Table 1 to Table 4, we observed that the results of the proposed system are better than the traditional IR system based on a Vector Space Model. The best results are summarized in Table 5. Using a lemmatizer on the list of articles improves the performance by approximately 10% from the previous work.

The thesauri, in addition to the lemmatizer, were used with the purpose of improving the search results; however, we did not obtain significant improvement. Furthermore, combining both strategies led to a decrease of performance with regard to only lemmatizing. The thesaurus expands the terms of the query. Usually it augments it with 9 or more terms per content word. It is possible that 9 terms are so many, so that we should experiment by using only 2 or 3 of them. In addition, the thesauri we are using (both manual and distributional) are general, as they are based in Anaya Dictionary and Encarta Encyclopaedia, respectively. So is then, that the query is expanded with terms not necessarily related with the context, creating confusion.

In general, dividing the query in different ways, as well as traversing the nodes in one way or another does not affect very much the performance of the system. This is a good effect for the model, since it shows that finding answers within the information contained there is not highly sensible to a particular way of using the graph, *i.e.*, the information is contained mainly in the way the structure is created, and not in the way it is traversed.

There is still room for improvement, as a future work we propose using a distributional thesaurus based on the same corpus of legal documents, as well as using different degrees of query expansion and using different similarity measures other than TF·IDF.

References

1. Hirschman, L., Gaizauskas, R.: Natural Language Question Answering: The View From Here. Natural Language Engineering 7(4), 275–300 (2001)
2. Hoojung, C., Song, Y.-I., Han, K.-S., Yoon, D.-S., Lee, J.-Y., Rim, H.-C.: A Practical QA System in Restricted Domains. In: Workshop on Question Answering in Restricted Domains. 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004), Barcelona, Spain, pp. 39–45 (2004)
3. Erik, T., Sang, K., Bouma, G., de Rijke, M.: Developing Offline Strategies for Answering Medical Questions. In: Workshop on Question Answering in Restricted Domains. 20th National Conference on Artificial Intelligence (AAAI 2005), Pittsburgh, PA, pp. 41–45 (2005)
4. Fabio, R., Dowdall, J., Schneider, G.: Answering questions in the genomics domain. In: Proceedings of the ACL 2004 Workshop on Question Answering in Restricted Domains, Barcelona, Spain, pp. 46–53 (2004)
5. Niu, Y., Graeme, H.: Analysis of Semantic Classes in Medical Text for Question Answering. In: Workshop on Question Answering in Restricted Domains. 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004), Barcelona, Spain, pp. 54–61 (2004)
6. Zhuo, Z., Da Sylva, L., Davidson, C., Lizarralde, G., Nie, J.-Y.: Domain-Specific QA for the Construction Sector. In: Workshop of IR4QA: Information Retrieval for Question Answering, 27th ACM-SIGIR, Sheffield (July 2004)
7. Paulo, Q., Rodrigues, I.P.: A question-answering system for Portuguese juridical documents. In: Proceedings of the 10th international conference on Artificial intelligence and law. International Conference on Artificial Intelligence and Law, Bologna, Italy, pp. 256–257 (2005)
8. Paulo, Q., Rodrigues, I.P.: A collaborative legal information retrieval system using dynamic logic programming. In: Proceedings of the 7th International Conference on Artificial Intelligence and Law, Oslo, Norway, pp. 190–191 (1999)
9. Doan-Nguyen, H., Kosseim, L.: The problem of precision in restricted-domain question-answering. Some proposed methods of improvement. In: Workshop on Question Answering in Restricted Domains. 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004), Barcelona, Spain, pp. 8–15 (2004)

10. Diekema Anne, R., Yilmazel, O., Liddy, E.D.: Evaluation of restricted domain question-answering systems. In: Workshop on Question Answering in Restricted Domains. 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004), Barcelona, Spain, pp. 2–7 (2004)
11. Rada, M.: Random Walks on Text Structures. In: Gelbukh, A. (ed.) CICLing 2006. LNCS, vol. 3878. Springer, Heidelberg (2006)
12. Manning Christopher, D., Schutze, H.: Foundations of Statistical Natural Language processing. MIT Press, Cambridge (1999)
13. Salton, G., Wong, A., Yang, C.S.: A vector Space Model for Automatic Indexing. Information Retrieval and Language Processing (1975)
14. Dan, M., Clark, C., Harabagiu, S., Maiorano, S.: COGEX: a logic prover for question answering. In: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, Edmonton, Canada, vol. 1, pp. 87–93 (2003)
15. Rila, M., Tokunaga, T., Tanaka, H.: Query expansion using heterogeneus thesauri. Information Processing and Management 36, 361–378 (2000)
16. Pizzato, L.A.S., de Lima, V.L.S.: Evaluation of a thesaurus-based query expansion technique. In: Mamede, N.J., Baptista, J., Trancoso, I., Nunes, M.d.G.V. (eds.) PROPOR 2003. LNCS, vol. 2721, pp. 251–258. Springer, Heidelberg (2003)
17. Calvo, H., Gelbukh, A., Kilgarriff, A.: Distributional thesaurus versus wordNet: A comparison of backoff techniques for unsupervised PP attachment. In: Gelbukh, A. (ed.) CICLing 2005. LNCS, vol. 3406, pp. 177–188. Springer, Heidelberg (2005)
18. Biblioteca de Consulta Microsoft Encarta 2004, Microsoft Corporation (1994–2004)
19. Lin, D.: An information-theoretic measure of similarity. In: Proceedings of ICML 1998, pp. 296–304 (1998)
20. Alfredo, M., Calvo, H., Gelbukh, A.: Using Graphs for Shallow Question Answering on Legal Documents. In: Gelbukh, A., Morales, E.F. (eds.) MICAI 2008. LNCS, vol. 5317, pp. 165–173. Springer, Heidelberg (2008)
21. Lázaro Carreter, F. (ed.): Diccionario Anaya de la Lengua, Vox (1991)