# Stemming and n-grams in Spanish: An Evaluation of Their Impact on Information Retrieval

Carlos G. Figuerola, Raquel Gómez, Eva López de San Román

Universidad de Salamanca

Abstract:

At some stage, most of the models and techniques implemented in IR use frequency counts of the terms appearing in documents and in queries. However, many words, since they are derived from the same stem, have very close semantic contents.  This makes a grouping of such variants under a single term advisable.  Otherwise, dispersal occurs in the calculation of frequency of these terms, and it also becomes difficult to compare queries and documents.  On the other hand, there are notable differences between different languages in the way of forming derivatives and inflected forms, so that the application of specific techniques can produce unequal results according to the language of the documents and queries.  A description is given of the tests carried out for documents in Spanish, which involved some stemming techniques widely used in English, as well as the application of n-grams, and the results are compared.

Most of the models and techniques employed in IR at some stage use frequency counts of the terms that appear in documents and queries. However, in this context, the concept of term is not exactly equivalent to that of word. Leaving aside the matter of so-called empty words, which cannot be considered terms as such, we have the case of words derived from the same stem, which can be said to have a very close semantic content [1].

The possible variations of the derivatives, together with inflected forms, changes in gender and number, etc., make the grouping of these variants under a single term advisable. Otherwise, dispersal in the calculation of frequency of these terms occurs, and it is difficult to compare queries and documents [2].

On the other hand, if this grouping does not occur, the comparison between a query and the documents of a collection becomes problematic. Somehow the programmes that are to solve this query must identify inflected forms or derivatives –which may be different in the query and in the document– as similar and corresponding to the same stem.

Therefore, it is necessary to incorporate into information retrieval systems a mechanism that makes it possible to undertake the standardization of different words, in the sense of representing the different forms of one same stem under one same form which may appear in documents and queries. This operation is generically known as “stemming”, in that it is a matter of automatically obtaining the stem corresponding to each word that may appear in documents and queries.

The very concept of stem can be approached in different ways. Although it seems evident that the derivation, and even the mere inflexion, of a word modifies its semantic content, there is no clear line that makes it possible to delimit to what extent we are dealing with forms corresponding to one same stem, or whether it is a clearly differentiated term [3]. Naturally, this is directly related to the specific objective that stemming pursues. In our case, this objective consists of improving the performance of IR systems, but for other types of applications the criteria do not have to be the same.

Thus, a linguist will admit that a change in gender or number, for example, is perfectly acceptable; in this way, 'catalogue' and 'catalogues' clearly correspond to the same stem. However, something different occurs with 'catalogue' and 'cataloguing', for example, although it is evident that they are different words, even belonging to different grammatical categories. For the purposes of retrieval it seems reasonable to suppose that if someone makes a query related to 'catalogue' they should obtain documents in which the word 'cataloguing' appears.

This question has been posed in diverse ways, from a simple stripping to the application of rather more sophisticated algorithms. Among the more well-known contributions we find the algorithm proposed by Lovin in 1968 [4], which, to some extent, is the basis of subsequent algorithms and proposals, such as those of Dawson [5], Porter [2] and Paice [6].

The results of the different forms of stemming, however, are irregular. Thus, they have been abundantly applied to texts in English with satisfactory results. With other, more morphologically complex languages, such as those derived from Latin, it is quite a different matter. On the one hand, there has been generally less IR work done in these languages and on the other hand, the application of stemming algorithms requires the implementation of considerable linguistic knowledge, which is not always available. In any case, it is possible to find proposals and algorithms for specific languages, among which are Latin itself, despite its being a dead language [7], Malay [8], French [9], [10] or Arabic [11].

## N-grams

An alternative to the difficulties posed by stemming is the use of n-grams in place of terms. N-grams are a kind of window of n characters in size, which progressively slide along the words of documents. In general, words preceded and followed by a blank space are considered [12]. Thus, for an n = 3 size, for example, the word ‘biblioteca’ would give rise to the following n-grams: ‘_bi’, ‘bib’, ‘ibl’, ‘bli’, ‘lio’, ‘iot’, ‘ote’, ‘tec’, ‘eca’, ‘ca_’ (where the symbol _ equals a blank space). Words with the same root, but with different suffixes, would give a certain number of equal n-grams plus some different ones –those corresponding to the suffixes. This partial coincidence of the n-grams produced would serve to establish a similarity between both words, despite their not being exactly the same.

For the same reason, the extraction of n-grams can attenuate problems derived from typographical errors and spelling mistakes [13]. The n-grams have been applied profusely in tasks related to information retrieval. For the case in question, the handling of derivatives and inflected forms of words, we must mention the work of Adamson and Boreham [14], as well as those of Lennon and colleagues [15], and those of Cavnar [16], Damasek [17] and Huffman [18].

## IR and Stemming in Spanish

Research in IR with documents in Spanish is not particularly abundant. Probably the most well-known studies are those carried out in the TREC conference setting [19] especially Trec-3 [20] and Trec-4 [21]. These conferences, as is well known, included collections of documents and queries in Spanish and several of the participants reported on their experiences with this language. On the whole, their specific linguistic knowledge of Spanish is scanty, which makes it difficult to evaluate the effectiveness of stemming from the point of view of retrieval, or even stemming itself.

In the main, all of them apply the same techniques and algorithms that are used for collections in English; the difference lies, naturally, in the lists of suffixes and, in some cases, in additional rules that some algorithms employ. Thus, in the work of the Cornell University team in TREC-3 [22], for example, a list of only six suffixes was applied, as well as a clearly surprising rule, that of changing the final z of some words to a c. A detailed analysis of the works that applied stemming techniques to Spanish can be found in [23], but it can be said that, in general, they suffer from a lack of specific linguistic knowledge.

For this reason we felt that it would be interesting to verify experimentally the impact of stemming on IR with documents and queries in Spanish. We thus applied a standard algorithm, that of Porter [2], but with sufficient and correct Spanish suffixes. We also applied a system of n-grams of different sizes and then compared the results obtained.

## The Document Collection

The document collection Datathèke was used in the experiments. This is a small collection of slightly more than 1000 documents in Spanish, all of which are abstracts of articles on Library and Information Science. The collection also includes a battery of 15 queries (also in Spanish), for which we have the corresponding estimations of relevance calculated manually, i.e. we know which of the documents are relevant to each of the queries.

## The Retrieval Model

In order to carry out retrieval in the different experiments a system based on the classic vector model was used, the weight of the terms being calculated according to the term frequency x IDF standard scheme [24]. Similarity between queries and documents was calculated by the well-known cosine formula [25]. Furthermore, the system also has a standard list of empty words (approx. 400), prepared *a priori*.

## Stemming

For stemming, a list of some 300 suffixes and their allomorphs was made, most of which can be applied to nouns, adjectives, and verbal forms that can function as nouns or adjectives. The suffixes and their allomorphs were obtained from the Dictionary of the Real Academia Española [26] and of the Dictionary of M. Moliner [27], considered like the sources most authorized in this field. Basically, there are two types of suffixes: the flexives, that are used to express variations of gender, number, tense and person in a word; and the derivatives, that produce different words, but generally strongly related semantically [28].

The flexives suffixes are applied of enough way uniform in names and adjectives, but not in the irregular verbal forms. The derivatives suffixes present several problems, since they stick to the root of irregular way. A same suffix can be added to a root eliminating one or more letters by that root (for example _URA, with the ALTO (*high*) word produces ALTURA (*height*) and/or adding intermediate letters (for example, COSER (*to sew*) + _URA = COSTURA (*sewing*)). But, frequently, the modifications are more complex and difficult to treat; to follow with the same suffix that has served to us as example, we consider that CALOR (*heat*) +_URA produces CALENTURA (*warmth, fever*), or that ABRIR (*to open*)+ _URA can produce APERTURA (*opening*), or that LEER (*to read*) + _URA originates LECTURA (*reading*). Sometimes, which seems a same suffix, it can undergo trasformations based on the root with which it goes. Let us consider, for example, suffix _ANZA, like in CONFIANZA (*confidence*) (= CONFIAR + ANZA). With another different root it is possible to be turned _ANCIA (for example TOLERAR (*to tolerate*)+ ANZA = TOLERANCIA).

The suffixes corresponding to inflected forms of verbs were deliberately omitted. In this sense, the great abundance of verbal forms and irregular verbs in Spanish must be taken into account, on the one hand, and, on the other, the assumedly slight importance –from the point of view of retrieval- of the verbal predicates [29].

Porter’s algorithm, working with the aforementioned suffixes, was implemented, with the addition of rules for detecting plurals and obtaining singular forms. In Spanish, the forming of the plural follows simple and fairly stable rules, i.e., there are very few exceptions. Thus, if the word ends in an unstressed vowel, the plural is formed by adding ‘s’, whereas if it ends in a consonant or a stressed vowel, the plural is formed by adding ‘es’.

The battery of queries was applied without stemming, with stemming by Porter’s algorithm and the detection of plurals and only with detection of plurals. In all three cases the empty words were previously eliminated; for application of Porter’s algorithm different minimum root sizes were tried out. In all three cases, accuracy and thoroughness were estimated and the results are given in Graph 1.

The results show a clear inferiority of non-stemming, which was, however, foreseeable. Nevertheless, there is no appreciable difference between the application of Porter’s algorithm plus the detection of plurals and the detection of plurals alone. This leads us to several considerations; in the first place, it must be taken into account that in Spanish suffixes can be added in many different ways, often altering the root of the word and even the suffix itself. But, moreover, this occurs with many variations, so that one same suffix can function differently according to the root that it accompanies; even so, the possible rule to be applied must contemplate many exceptions and particular cases.

This leads to algorithms such as Porter’s being considered unsuitable for Spanish, and also reveals the need to formalise a great deal of complex linguistic knowledge. Unfortunately, linguistic tools for Spanish are scarce and require greater development.

## N-Grams in Spanish

Using the same collection of documents and queries, and the same retrieval model, n-grams of different sizes were extracted and used as terms in the vectors. As much documents as queries were reduced to n-gramas. Previously, the empty words had been eliminated. With respect to the size of n, previous studies have experimented with n=3 and n=4, with unequal results [16]. In our case, for these sizes of n the results obtained were frankly discouraging. However, working with n=5, n=6, and even n=7, much better results were obtained, as can be seen in Graph 2.

The unusually large size of n is notable, given that the studies described in the literature available to us use a maximum size of 4, and even affirm that larger sizes bring about a drop in effectiveness. The reason for this disparity can be found in the fact that a large part of the previous studies were performed on collections in English and that application to collections in Spanish was performed following the same patterns, without taking into account the peculiarities and differences in the languages. The different number of characters in the words of one language and the other probably have an effect on this. Thus, in Spanish, there are frequently longer words than in English. By way of example, the average length of the words in the collection used in this study is 7.88 characters (without counting empty words, which tend to be shorter).

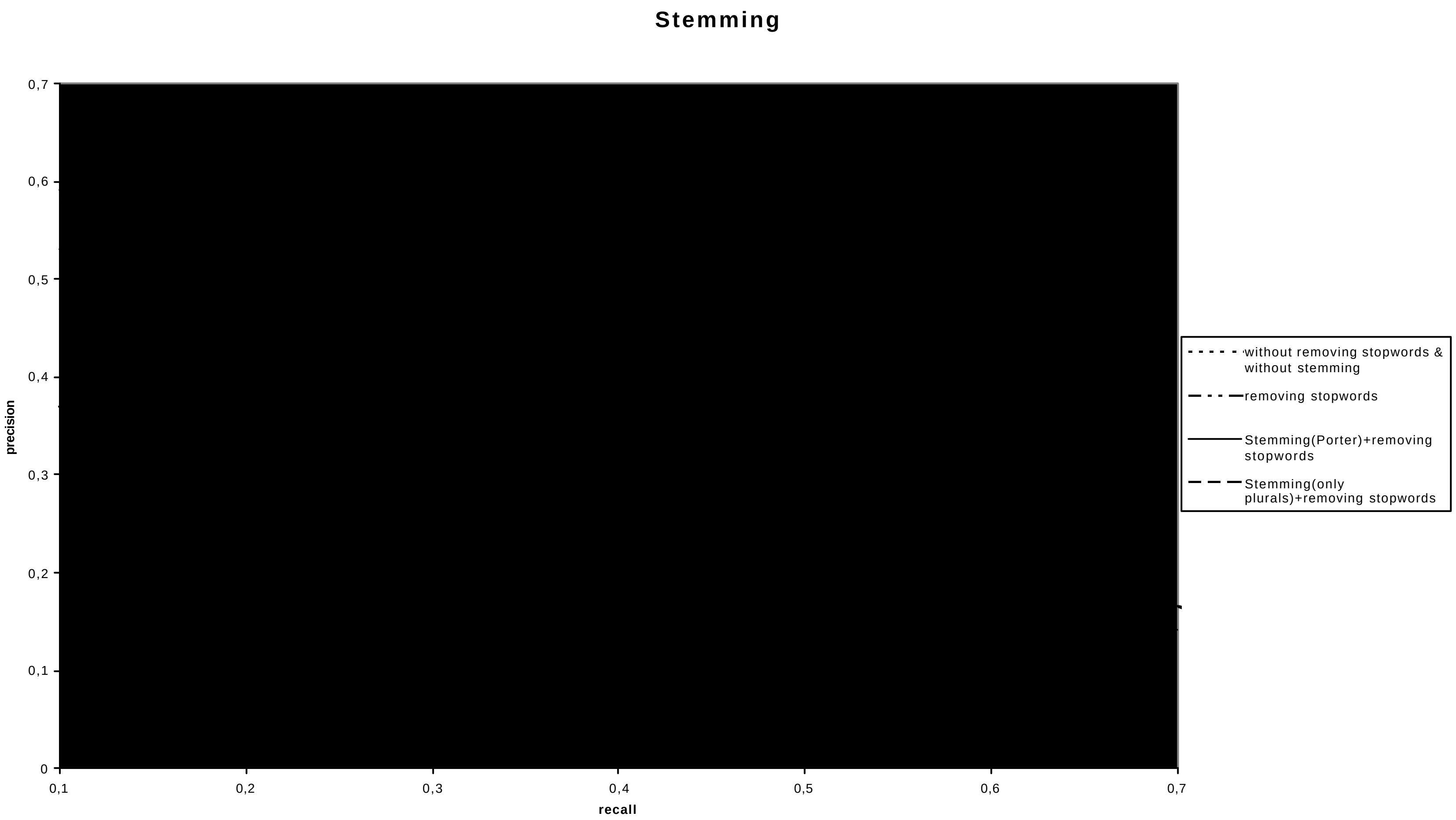
## Conclusions

The results of the retrieval tests, measured in terms of precision and recall, seem quite clear. As regards n-grams, two basic aspects should be underlined: on the one hand, the size of n, which seems to work better with values of 6 and 7. Despite the fact that it is usual to work with smaller values, the results obtained with n=4 are clearly worse than those obtained with vales of n=6 or n=7. Moreover, it seems that values of above 7 tend to worsen rather than improve the results; this is the case with n=8.

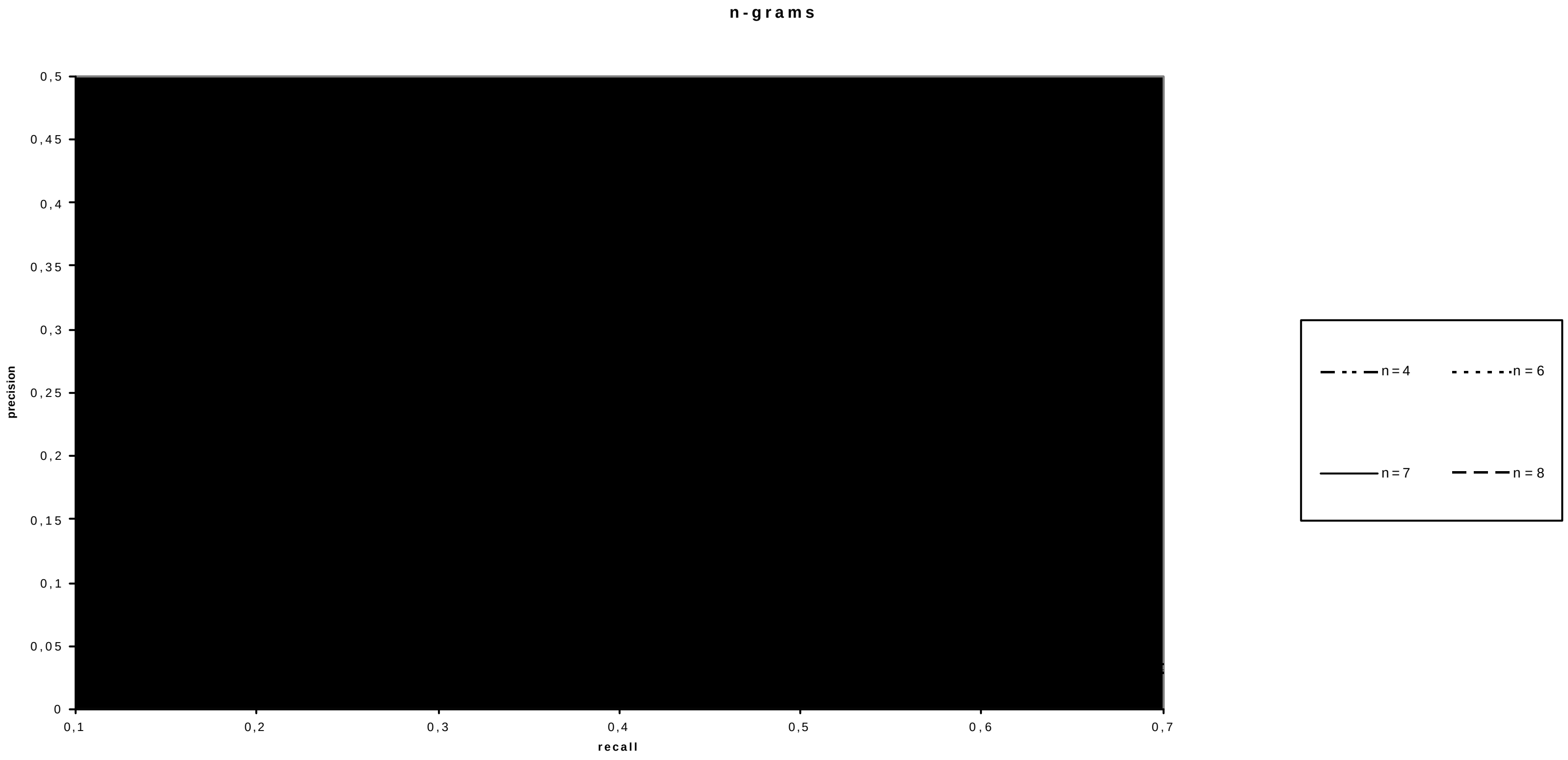
The other aspect related to the n-grams is that of the poor results, from any point of view, obtained with this technique. Note that the best result obtained with n-grams is worse than that obtained by retrieval with complete terms, without the prior elimination of empty words or the use of any kind of stemming, and in this sense the differences seem conclusive.

With respect to stemming, it seems clear that it improves the results in retrieval. However, it should be pointed out that, as regards results, there is scarce difference between stemming by applying Porter's algorithm and the mere elimination of plurals by their conversion to the corresponding singular form; in fact, the results of the latter technique can even be considered somewhat better. Thus the conclusion, regarding this matter, seems obvious: the methods based on Porter's algorithm and similar techniques are unsuitable for documents in Spanish; the linguistic peculiarities of each language play an important role and it is necessary to implement techniques that incorporate the appropriate linguistic knowledge.

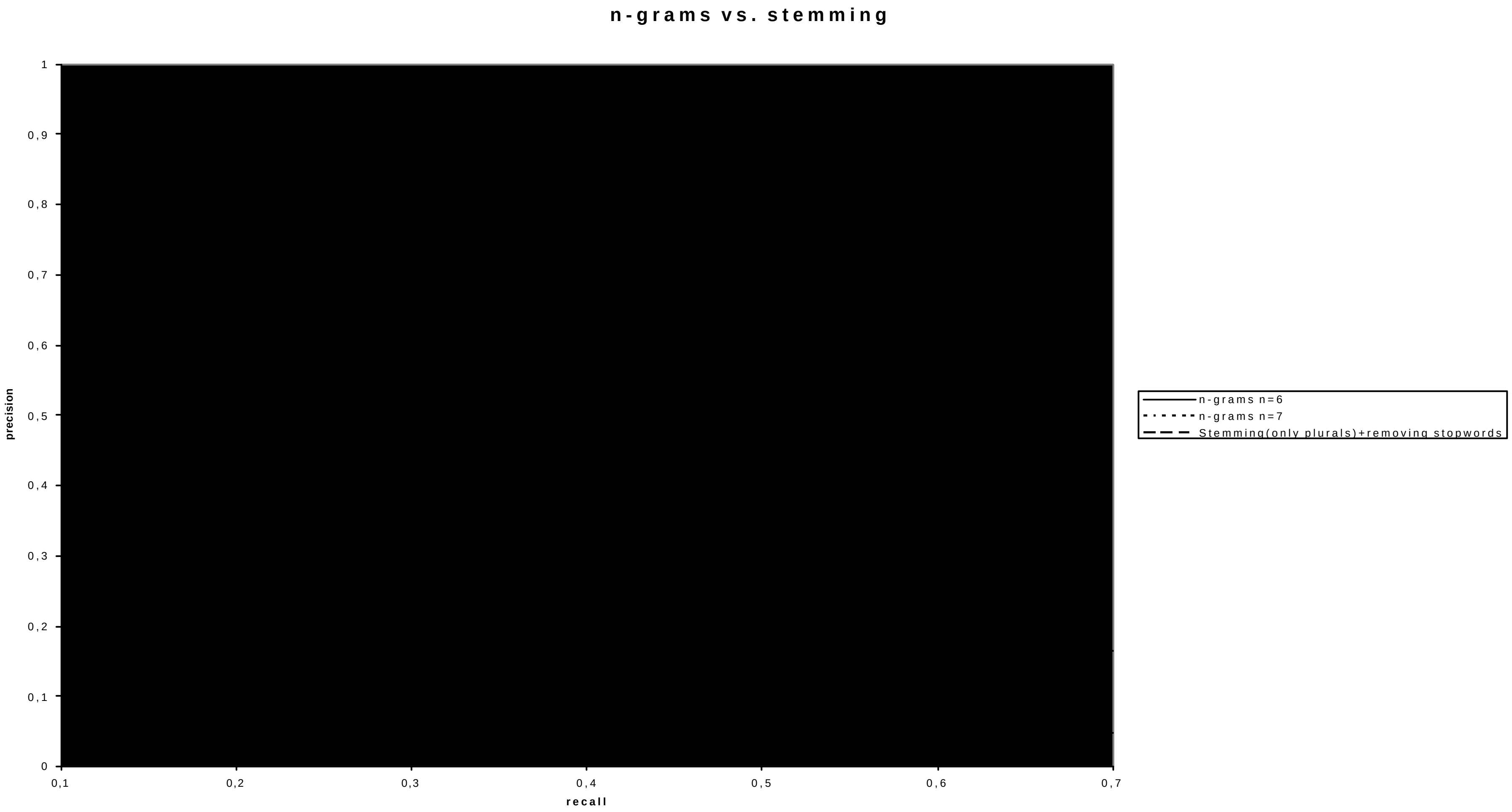
This linguistic knowledge, on the other hand, can be somewhat richer and more complex than one might at first think. In other words, it is probably not enough to handle a simple list of suffixes and endings, no matter how long it is.  In Spanish, as in other morphologically complex languages, the endings or suffixes are not simply tacked on to the root, but are added in different ways, by modifying the suffix itself, and also the root to which it is added.


Stemming
precision
0,7
0,6
0,5
0,4
0,3
0,2
0,1
0
0,1
0,2
0,3
0,4
0,5
0,6
0,7
recall
without removing stopwords & without stemming
removing stopwords
Stemming(Porter)+removing stopwords
Stemming(only plurals)+removing stopwords

Graph. 1


n-grams
precision
0,5
0,45
0,4
0,35
0,3
0,25
0,2
0,15
0,1
0,05
0
0,1
0,2
0,3
0,4
0,5
0,6
0,7
recall
n = 4
n = 6
n = 7
n = 8

Graph. 2


n-grams vs. stemming
precision
1
0,9
0,8
0,7
0,6
0,5
0,4
0,3
0,2
0,1
0
0,1
0,2
0,3
0,4
0,5
0,6
0,7
recall
n-grams n=6
n-grams n=7
Stemming(only plurals)+removing stopwords

**Graph. 3**

REFERENCES:

[1] Salton, G. : *Automatic Information Organization and Retrieval* (McGraw-Hill, New York, 1968)

[2] Porter, M. F. (1980): An algorithm for suffix stripping, *Program (London)*, 14(3) (1980), 130-137

[3] Paice, C. D.: Method for evaluation of stemming algorithms based on error counting, *Journal of the American Society for Information Science*, 47(8) (1996), 632-649

[4] Lovin, J.B.: Development of a Stemming Algorithm, *Mechanical Translations and Computational Linguistics*, 11(1-2)(1968), 22-31

[5] Dawson, J.: Suffix Removal and Word Conflation, *ALLC Bulletin*, 1974, 33-46

[6] Paice, C. : Another Stemmer, *ACM SIGIR Forum*, 24(3) (1990), 56-61

[7] Schinke, R., Robertson, A., Willett, P., Greengrass, M. : A stemming algorithm for Latin text databases, *Journal of Documentation*, 52, (2) (1996) 172-187

[8] Ahamad, F., Yussof, M. and Sembok, M. T. : Experiments with a Stemming Algorithm for Malay Words, *Journal of the American Society for Information Science*, 47, (12) (1996) 909-918

[9] Savoy, J. : Stemming of French words based on grammatical categories, *Journal of the American Society for Information Science*, 44(1) (1993), 1-9

[10] Savoy, J.: A Stemming Procedure and Stopword List for General French Corpora, *Journal of the American Society for Information Science*, 50(10) (1999), 944-952

[11] Abu-Salem, H.; Al –Omari, M. and Evens, M.W. : Stemming Methodologies Over Individual Query Words for an Arabian Information Retrieval System, *Journal of American Society for Information Science*, 50(6) (1999) 524-529

[12] Robertson, A.M. and Willet, P. : Applications of n-grams in textual information systems, *Journal of Documentation*, 54(1) (1998), 48-69

[13] Pollock, J.J. and Zamora, A. : System design for detection and correction of spelling errors in scientifc and scholarly text, *Journal of American Society for Information Science*, 35 (1984)104-109

[14] Adamson, G.W. and Boreham, J. : The use of an association measure based on character structure to identify semantically related pairs of words and document titles, *Information Storage and Retrieval*, 10 (1974) 253-260.

[15] Lennon, M. , Peirce, D.S., Tarry, B.D. and Willett, P. : An evaluation of some conflation algorithms for information retrieval, *Journal of Information Science*, 3 (1981)177-183

[16] Cavnar, W.B.: Using An N-Gram Based Document Representation With A Vector Processing Retrieval Model, *TREC-3*, Special NIST Pub. N. 500-226, Gaittersburg, Maryland, 1994 [http://trec.nist.gov/pubs/trec3/papers/cavnar_ngram_94.ps]

[17] Damashek, M. : Gauging similarity with n-grams: language independent categorisation of text, *Science*, 267 (1995) 843-848

[18] Huffman, S. : Acquaintance: Language-Independent Document Categorization by N-Grams, *TREC-4*, Special NIST Pub. N. 500-236, Gaittersburg, Maryland, 1995, [http://trec.nist.gov/pubs/trec4/papers/nsa.ps]

[19] Harman, D. : The TREC Conferences, *Proceedings HIM'95 (Hypertext-Information Retrieval-Multimedia)*, Konstanz (1995), 9-23

[20] Harman, D.K. (ed.): *Overview of the Third Text Retrieval Conference (TREC-3)*, Special NIST Pub. N. 500-226, Gaittersburg, Maryland, 1994 [http://trec.nist.gov/pubs/trec3/t3_proceedings.html]

[21] Harman, D.K. (ed.): *The Fourth Text Retrieval Conference (TREC-4)*, Special NIST Pub. N. 500-236, Gaittersburg, Maryland, 1995 [http://trec.nist.gov/pubs/trec4/t4_proceedings.html]

[22] Buckley, C., Salton, G., Allan, J. and Singhal, A. : Automatic Query Expansion Using SMART: TREC3, *TREC-3*, Special NIST

Pub. N. 500-226, Gaittersburg, Maryland, 1994 [http://trec.nist.gov/pubs/trec3/papers/cornell.new.ps]

[23] Gómez Díaz, R. : *La Recuperación de Información en español: evaluación del efecto de sus peculiaridades lingüísticas* (Unpublished paper at Universidad de Salamanca, Salamanca, 1998)

[24] Harman, D. : Ranking Algorithms. In Frakes, W.B. and Baeza-Yates, R. (ed.), *Information Retrieval. Data Structures and Algorithms*, Prentice Hall, Upper Saddle River, NJ, 1992, 363-392

[25] Salton, G. and McGill, M.J.: *Introduction to modern Information Retrieval* (McGraw-Hill, New York, 1983)

[26] Real Academia Española: Diccionario de la lengua española, Madrid, 1996

[27] Moliner, M.: Diccionario de uso del español, Madrid, 1991

[28] Pérez Lagos, M.F.: Formación de palabras, la composición culta en los diccionarios, Salamanca, 1996

[29] Rodríguez Muñoz, J. V. y Gil Leiva, I. : Análisis de los descriptores de diferentes áreas del conocimiento indizadas en bases de datos del CSIC. Aplicación de la indización automática , *Revista Española de Documentación Científica*, 20(2) (1997), 150-160