Spanish Monolingual Track: the Impact of Stemming on Retrieval

Carlos G. Figuerola, Raquel Gómez, Angel F. Zazo Rodríguez, and José Luis Alonso Berrocal

Universidad de Salamanca
c/ Francisco de Vitoria, 6-16, 37008 Salamanca, Spain
[figue|afzazo|berrocal|rgomez]@usal.es

Abstract. Most of the techniques used in Information Retrieval rely on the identification of terms from queries and documents, as much to carry out calculations based on the frequencies of these terms as to carry out comparisons between documents and queries. Terms coming from the same stem, either by morphological inflection or through derivation, can be presumed to have semantic proximity. The conflation of these words to a common form can produce improvements in retrieval. The stemming mechanisms used depend directly on each language. In this paper a stemmer for Spanish and the tests conducted by applying it to the CLEF Spanish document collection are described, and the results are discussed.

1 Introduction

At one time or another, most of the models and techniques employed in Information Retrieval use frequency counts of the terms appearing in documents and queries. The concept of term in this context, however, is not exactly the same as that of word. Leaving to one side the matter of so-called stop words, which cannot be considered terms as such, we have the case of words derived from the same stem, to which can be attributed a very close semantic content [1]. The possible variations of the derivatives, together with their inflections, alterations in gender and number, etc., make it advisable to group these variants under one term. If this is not done, a dispersion in the calculation of the frequency of such terms occurs and difficulty ensues in the comparison of queries and documents [2].

Programs that process query must be able to identify inflections and derivatives -which may be different in the query and the documents- as similar and as corresponding to the same stem. Stemming, as a way of standardising the representation of the terms with which Information Retrieval systems operate, is an attempt to solve these problems.

However, the effectiveness of stemming has been the object of discussion, probably beginning with the work of Harman [3], who, after trying several algorithms (for English), concluded that none of them increased effectiveness in

retrieval. Subsequent work [4] pointed out that stemming is effective as a function of the morphological complexity of the language being used, while Krovetz [5] found that stemming improves recall and even precision when documents and queries are short.

2 Previous Work

Stemming applied to Information Retrieval has been tried in several ways, from succinct stripping to the application of much sophisticated algorithms. Far studies began in the 1960s with the aim of reducing the size of indices [6] and, apart from being a way of standardising terms, stemming can also be seen as a means to expand queries by adding inflections or derivatives of the words to documents and queries.

One of the best known contributions is the algorithm proposed by Lovin in 1968 [7], which is in some sense the basis of subsequent algorithms and proposals, such as those of Dawson [8], Porter [2] and Paice [9]. Although much of the work reported is directed towards use with documents in English, it is possible to find proposals and algorithms for other languages, among them Latin, in spite of its being a dead language [10], Malaysian [11], French [12], [13], Arabic [14], Dutch [15], [16], Slovene [4] and Greek [17].

Various stemming mechanisms have been applied to Information Retrieval operations on Spanish texts in some of the TREC conferences (Text Retrieval Conference) [18]. In general, these applications consisted in using the same algorithms as for English, but with suffixes and rules for Spanish. Regardless of the algorithms applied, and of their adaptation to Spanish, the linguistic knowledge used (lists of suffixes, rules of application, etc.), was quite poor [19].

From the language processing perspective, in recent years several stemmers and morphological analysers for Spanish have been developed, including the COES tools [20], made available to the public at *http://www.datsi.fi.upm.es/~coes/* under GNU licensing; the morpho-syntax analyser MACO+ [21] (*http://nipadio.lsi.upc.es/cgi-bin/demo/demo.pl*) or the FLANOM / FLAVER stemmers [22], [23] (*http://protos.dis.ulpgc.es/*). However, we are unaware of any experimental results of the application of these tools to Information Retrieval.

On the other hand, on several occasions the use of n-grams has been proposed to obviate the problem posed by inflectional and derivational variation [24]. In previous work, however, we were able to verify the scant effectiveness of this mechanism from the point of view of Information Retrieval [25], as well as the inadequacy of the well-known Porter algorithm for languages such as Spanish.

3 The Stemmer

The basis of our stemmer consists of a finite states machine that attempts to represent the modifications undergone by a stem when a certain suffix is attached or added to it. There is thus an instance of this automaton for each suffix contemplated: each of these implies a series of rules expressing how that suffix is

incorporated into the stem. Since, for the same suffix, at times there may be a large number of variants and exceptions, the resulting automaton can be quite complex.

Thus, in order to stem a word, the longest suffix coinciding with the end of this word is sought and the corresponding automaton is formed with the rules for that suffix. The network of this automaton is searched with the word to be stemmed and the string obtained in the terminal node of the automaton is contrasted with a dictionary of stems. If the chain obtained is found in the dictionary the stem is considered to be correct.

Taking into account that the transformations may occasionally overlap, adding more than one suffix, the process is repeated recursively until the correct stem is found. If, once the possibilities are exhausted, none of the terminal strings obtained are found in the dictionary of stems, it is deduced that either the word can be considered as standardised in itself, or else it is a case not foreseen by the stemmer.

This last instance may mean the following:

1. the word has a suffix that is not included in the list of suffixes of the stemmer
2. the suffix is added in a way that was not predicted by the rules incorporated in the knowledge base
3. the stem is not in the dictionary of stems.

Thus, the stemmer to be subjected to a training process in which the results of stemming the words of a corpus are examined manually and the knowledge base of the stemmer is corrected when necessary.

We can distinguish between two classes of stemming: inflectional and derivative. Whereas the former has clear and defined limits, this does not occur with the latter. Moreover, the semantic distance between two different inflections of the same stem can in general be considered of little importance (for example, *libro* and *libros*), whereas the semantic difference between a stem and its derivatives may be great; for example, *sombra* (shade), *sombrilla* (parasol, sunshade), and even *sombrero* (hat).

Inflectional stemming should heed changes in gender and/or number for nouns and adjectives and changes in person, number, tense and mode for verbs. Treatment of nouns and adjectives is simple, since both changes in gender and in number follow simple rules. Exceptions to these rules exist but they there are few and can be treated individually. Verbs, however, are another case. Besides the great number of forms a verb can take, the main problem lies in the large amount of irregular verbs in Spanish. There may be more than 40,000 irregular verbs and any basic course of Spanish includes lists of 8 or 10 thousand irregular verbs. Fortunately, these can be grouped into approximately 80 different models, although they do not always strictly follow a given model and there are many exceptions.

In inflectional stemming there is another complex problem to be solved: the grammatical ambiguity of many words. A certain word ending in a certain suffix may pertain to different grammatical categories and, depending on which it

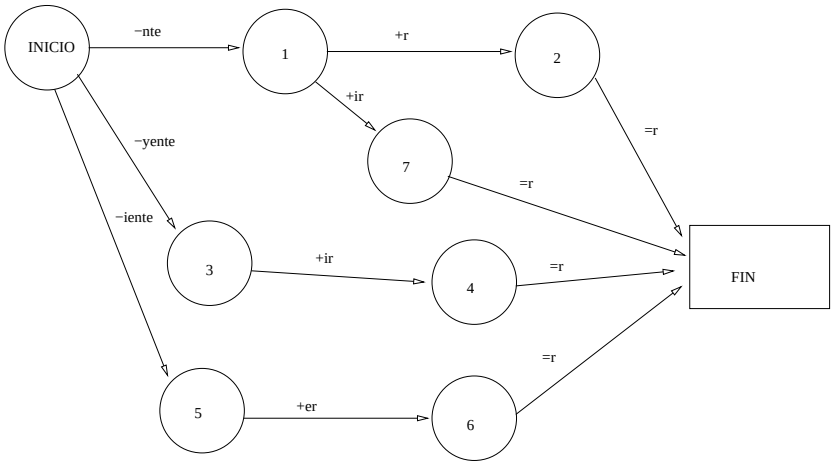

Fig. 1. Automaton for the suffix -*ente*

pertains to, the inflectional transformations it has undergone will be different and will in consequence have come from different stems. A simple example would be the word *colecciones*: it could be the plural of the noun *colección* (collection) or else the second person singular present subjunctive of *coleccionar* (to collect), and would thus give rise to two different stems.

The way to solve this ambiguity could lie in resorting to the specific context of the word and determining its grammatical category, in order to then choose the right stem. Our stemmer cannot yet resolve this ambiguity. However, one should take into account that some forms are more frequent than others; a verb in subjunctive mode is much more infrequent than a noun, and even more so in journalistic texts such as the ones we have dealt with.

For the moment, until we manage to solve this ambiguity, our stemmer chooses the most frequent stems; this necessarily introduces an element of error, but since it always applies the same stem, the error is always less than it would be without stemming. Furthermore, derivation produces a much higher number of forms based on one stem. Inflectional transformations can occur on any of these forms and therefore derivative stemming should be carried out after flexional stemming; for example, *libreros* (book-sellers) is a plural noun that should be reduced to singular in order to eliminate the suffix and end up with the stem *libro* (book).

4 The Impact of Stemming on Information Retrieval

The 50 queries of the CLEF Spanish monolingual collection were processed in three modalities: without stemming, applying inflectional stemming and applying inflectional plus derivational stemming. Obviously, the stemming was applied

both to documents and queries, and in all three cases stop words were eliminated previously, based on a standard list of 538 (articles, conjunctions, prepositions, etc.).

The algorithm is the same for both inflectional and derivative stemming. What changes, obviously, are the suffixes and rules of application, as well as the dictionary or list of stems to be used. For inflectional stemming the number of suffixes considered was 88, with a total of 2,700 rules of application. The dictionary of stems consists of 80,000 entries. For derivative stemming the number of suffixes is higher (since it is actually a matter of inflection plus derivatives): 230 with 3,692 rules of application. The dictionary or list of stems is much shorter: approximately 15,000 stems.

After eliminating stop words, the document collection produced a total of 36,573,577 words, with 353,868 unique words. Inflectional stemming reduced these 353,868 unique words to 284,645 stems; nevertheless, of these, 141,539 (almost half) were stems that appeared only once in the document collection. A simple glance shows that a good part of them correspond to typographical errors (which cannot be stemmed without previous detection and correction), as well as to proper names, acronyms, etc. Derivative stemming reduced the number of stems: the 353,868 unique words produced 252,494 single stems. Of these, 127,739 appeared only once in the document collection; most of them are typographical errors.

4.1 The Retrieval Model

To execute or solve the queries we used our own retrieval engine, Karpanta, [26], which is based on the well known vectorial model, defined by Salton some time ago [27]. The weights of the terms were calculated according to the usual scheme of *Frequency of term in the document* x *IDF*. IDF (Inverse Document Frequency) is an inverse function of the frequency of a term in the entire collection (understood as the number of documents in which it appears) [28]. The similarity between each document and each query was calculated using the formula of the cosine, as is usual in these cases [29].

Taking into account that our objective was to evaluate the effect of stemming, we did not consider it necessary to apply additional techniques such as feedback of queries [30], although the Karpanta retrieval system permits this. Actually, our intention was not so much to achieve the best results, but to measure the differences among the results obtained with each of the three modalities mentioned above.

5 Results

The results can be seen in the attached plot, and they are somewhat disappointing. The differences among the cases are scarce. Inflectional stemming produces about 7 % of improvement over unstemming. Derivative stemming is even a little bit worse than no stemming.

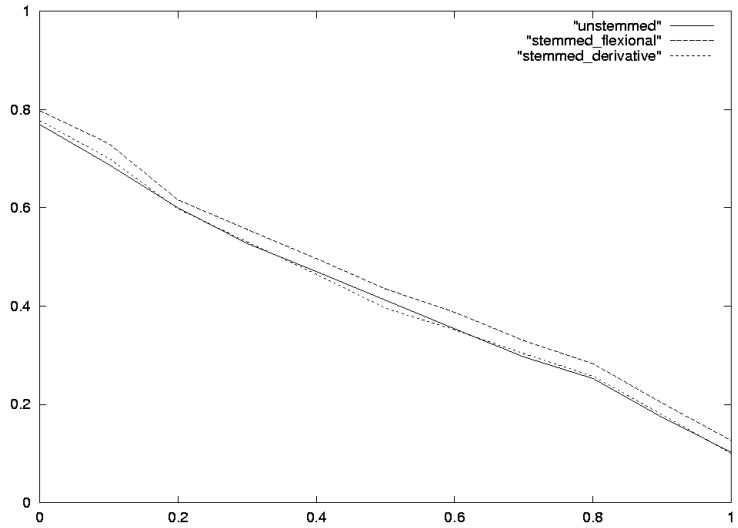

Fig. 2. Results of the official runs

This is somewhat surprising, because it differs considerably from the results obtained in previous tests with another document collection. In effect, both stemmers, inflectional and derivative, were tested with the Datathéke collection, obtaining significantly better results. Datathéke is a small homogeneous collection of 1074 documents in Spanish, consisting of scientific paper abstracts in Information and Library Science. Additionally, it has a set of 15 queries with corresponding relevance assestsments.

The test results with this small collection can be seen in Fig. 3, and they show a significant improvement using both stemmers, specially the inflectional one. Certainly, Datathéke is not a very representative collection, fundamentally due to its small size, but even the difference in results is surprising.

There are, nevertheless, some important differences between two collections, in addition to the size, that can help to understand the difference between the results obtaines. First, the document collection used in CLEF consists of agency news. It is known that journalistic texts are specially flat with regard to morphology and syntax; thus, smaller morphologic complexity could attenuate the effects of stemming. Additionally, agency newswires are typically even poorer in stylistic complexity, since they are not written to be published such as, but in the hope that newspapers editors of each newspaper process the news that finally gets published from them.

Secondly, the use of proper nouns in the queries. The queries of Datathéke hardly contain proper nouns in their formulation. Proper nouns cannot be stemmed, and several CLEF queries for Spanish are based on proper nouns. So, they can largely be solved on the basis of these proper nouns. In fact, there

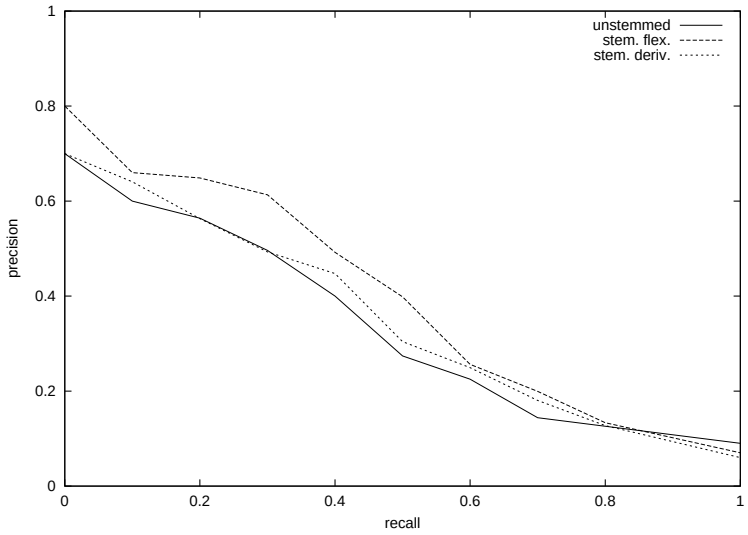

Fig. 3. Results of the *Datathéke* collection

are some queries in which results are practically obtained just using a simple substring search only with the proper nouns. For these queries the impact of stemming is, obviously, none. This happens with queries 46, 47, 48, 49, 50, 51, 66, 73, 79, 83, 88 and 89, and specially with the 66 and the 89. In these, for example, the same documents were retrieved by means of our Karpanta system (based on the vectorial model) and by means of a substring search with the terms *Letonia* and *Rusia* for query 66 and *Schneider* for 89.

6 Conclusions and Future Work

The stemming of terms from queries and documents in Spanish can be a means to improve the results in retrieval. This improvement depends on the morphological complexity of queries and documents. The inflectional stemming seems to produce better results than the derivational, since this one introduces excesive ambiguity in the stems.

For the future, we must to finish the stemmer, in particular with regards to the resolution of the ambiguity in the inflectional stems. This could be obtained by means of an examination of the context in which each term appears. In addition, we must decide whether the derivative stemming must be discarded definitively, or if a careful selection of the suffixes could be useful, discarding those that produce very semantically remote terms. Another possibility could be the use of thesauri or other linguistic resources to determine the relation between the stem and the derivative.

References

1. Hull, D.: Stemming algorithms: a case study for detailed evaluation. JASIS **47** (1996)
2. Porter, M.F.: An algorithm for suffix stripping. Program **14** (1980) 130–137
3. Harman, D.: How effective is suffixing? JASIS **42** (1991) 7–15
4. Popovic, M., Willet, P.: The effectiveness of stemming for natural-language access to slovene textual data. JASIS **43** (1992) 384–390
5. Krovetz, R.: Viewing morphology as an inference process. In: SIGIR 93. (1993) 191–203
6. Bell, C., Jones, K.P.: Toward everyday languaje information retrieval system via minicomputer. JASIS **30** (1979) 334–338
7. Lovins, J.B.: Development of a stemming algorithm. Mechanical Translation and Computational Linguistics **11** (1968) 22–31
8. Dawson, J.: Suffix removal and word conflation. ALLC bulletin **2** (1974) 33–46
9. Paice, C.D.: Another stemmer. In: SIGIR 90. (1990) 56–61
10. Schinke, R., Robertson, A., Willet, P., Greengrass, M.: A stemming algorithm for latin text databases. Journal of Documentation **52** (1996) 172–187
11. Ahmad, F., Yussof, M., Sembok, M.T.: Experiments with a stemming algorithm for malay words. JASIS **47** (1996) 909–918
12. Savoy, J.: Effectiveness of information retrieval systems used in a hypertext environment. Hypermedia **5** (1993) 23–46
13. Savoy, J.: A stemming procedure and stopword list for general french corpora. JASIS **50** (1999) 944–952
14. Abu-Salem, H., Al-Omari, M., Evens, M.W.: Stemming methodologies over individual queries words for an arabian information retrieval system. JASIS **50** (1999) 524–529
15. Kraaij, W., Pohlmann, R.: Porter's stemming algorithm for dutch. In Noordman, L.G.M., de Vroomen, W.A.M., eds.: Informatiewetenschap, Tilburg, STINFON (1994)
16. Kraaij, W., Pohlmann, R.: Viewing stemming as recall enhancement. In: SIGIR 96. (1996) 40–48
17. Kalamboukis, T.Z.: Suffix stripping with moderm greek. Program **29** (1995) 313–321
18. Harman, D.: The trec conferences. In: Proceedings of the HIM'95 (Hypertext-Information Retrieval-Multimedia). (1995) 9–23
19. Figuerola, C.G.: La investigación sobre recuparación de la información en español. In Gonzalo García, C. y García Yebra, V., ed.: Documentación, Terminología y Traducción, Madrid, Síntesis (2000) 73–82
20. Rodríguez, S., Carretero, J.: A formal approach to spanish morphology: the coes tools. In: XII Congreso de la SEPLN, Sevilla (1996) 118–126
21. Carmona, J., Cervell, S., Márquez, L., Martí, M., Padrón, L., Placer, R., Rodríguez, H., Taulé, M., Turmo, J.: An environment for morphosyntactic processing of unrestricted spanish text. In: Proceedings of the First International Conference on Language Resources and Evaluation (LREC'98), Granada, Spain (1998)
22. Santana, O., Pérez, J., Hernández, Z., Carreras, F., Rodríguez, G.: Flaver: Flexionador y lematizador automático de formas verbales. Lingüística Española Actual **XIX** (1997) 229–282
23. Santana, O., Pérez, J., Carreras, F., Duque, J., Hernández, Z., Rodríguez, G.: Flanom: Flexionador y lematizador automático de formas nominales. Lingüística Española Actual **XXI** (1999) 253–297

24. Robertson, A., Willet, P.: Applications of n-grams in textual information systems. Journal of Documentation **54** (1999) 28–47
25. Figuerola, C.G., Gómez, R., de San Román, E.L.: Stemming and n-grams in spanish: an evaluation of their impact on information retrieval. Journal of Information Science **26** (2000) 461–467
26. Figuerola, C.G., Berrocal, J.L.A., Rodríguez, A.F.Z.: Disseny d'un motor de recuperació d'informació per a ús experimental i educatiú = diseño de un motor de recuperación de información para uso experimental y educativo. BiD. textos universitaris de biblioteconomia i documentació **4** (2000)
27. Salton, G., McGill, M.: Introduction to Modern Information Retrieval. McGraw-Hill, New York (1983)
28. Harman, D.: Ranking algorithms. In: Information retrieval: data structures and algorithms, Upple Saddle River, NJ, Prentice-Hall (1992) 363–392
29. Salton, G.: Automatic Text Processing. Adisson-Wesley, Reading, MA (1989)
30. Harman, D. In: Relevance Feedback and Others Query Modification Techniques. Prentice-Hall, Upple Saddle River, NJ (1992)