# Testing Word Similarity: Language Independent Approach with Examples from Romance*

Mikhail Alexandrov[1], Xavier Blanco[2], and Pavel Makagonov[3]

[1]Center for Computing Research, National Polytechnic Institute (IPN), Mexico
dyner@cic.ipn.mx, dyner1950@mail.ru
[2]Department of French and Romance Philology, Autonomous University of Barcelona
Xavier.Blanco@uab.es
[3]Mixteca University of Technology, Mexico
mpp2003@inbox.ru

**Abstract.** Identification of words with the same basic meaning (stemming) has important applications in Information Retrieval, first of all for constructing word frequency lists Usual morphologically-based approaches (including the Porter stemmers) rely on language-dependent linguistic resources or knowledge, which causes problems when working with multilingual data and multithematic document collections We suggest several empirical formulae with easy to adjust parameters and demonstrate how to construct such formulae for a given language using an inductive method of model self-organization This method considers a set of models (formulae) of a given class and selects the best ones using training and test samples We describe the method and give detailed examples for French, Italian, Portuguese, and Spanish The formulae are examined on real domain-oriented document collections Our approach can be easily applied to other European languages

## 1 Introduction

Given a large text (or text corpus), we consider the task of grouping together the words having the same basic meaning, e.g., *sad*, *sadly*, *sadness*, *sadden*, *saddened*, etc.; this is usually referred as stemming The algorithm can make two types of errors: to join words that are not similar (false positive) or fail to join words which are similar (false negative). We can tolerate a certain amount of such errors because our main motivation is to improve performance of information retrieval rather than to do some precise linguistic analysis.

To group together different letter strings they are often labeled in a specific way, the strings with the same label being considered pertaining to the same group. As labels, standard dictionary forms (i.e., singular for nouns, indefinite infinitive for verbs, etc ) can be used; reducing a string to such a form is called lemmatization For lemmatization, morphology-based methods relying on morphological rules and a large morphological dictionary are usually used. They provide practically 100% accuracy on known words and good accuracy on the words absent in the dictionary [Gelbukh, 2003; Gelbukh and Sidorov, 2003, 2003]

* Work done under partial support of Mexican Government (CONACyT and CGEPI-IPN)

F. Meziane and E. Métais (Eds ): NLDB 2004, LNCS 3136, pp 229–241, 2004

Alternatively, stemming can be used to label strings [Porter, 1980]: the words are truncated to their stems, which often reflect their invariant meaning; e.g., *sad*, *sadly*, *sadness*, *sadden*, *saddened* are truncated to the same stem *sad-* Though this method is much simpler (since it usually relies only on lists of suffixes and suffix removal rules, but no dictionaries), it provides relatively low accuracy.

However, even such methods require large language-specific manually encoded rule sets with complicated structure. This becomes a problem in large-scale analysis of multilingual, multi-thematic document collections for which the effort required for manual compilation of suffix removal rules is prohibitive. In this paper we investigate the possibilities of using simple formulae with automatically trainable parameters.

Makagonov and Alexandrov [2002] have suggested an empirical method for testing word similarity and described a methodology for constructing empirical formulae based on the number of the coincident letters in the initial parts of the two words and the number of non-coincident letters in their final parts. The limited experiments on English texts with one of the formulae showed that for the documents from a specific domain it provides accuracy of 85% to 92%. The main advantage of this knowledge-poor approach is its simplicity and flexibility. It does not rely on any manually compiled morphological rules, dictionaries, suffix lists, or rule systems. To adapt the method to a new language or a new subject domain, only the parameters of the formula are to be automatically adjusted.

Still many important issues concerning application of the empirical formulae remain open. In this paper we give more detail on this method, investigate the sensibility of the formulae to different languages, and analyze the errors the method commits. In particular, we show that, similarly to other statistical characteristics [Gelbukh and Sidorov, 2001], the parameters of such formulae depend on language.

We considered here the Romance languages: French, Italian, Portuguese, and Spanish, in the domain widely discussed currently in the European Community: mortgage and crediting. To evaluate the accuracy of the results, we use characteristics both from mathematical statistics and from information retrieval.

## 2 The Problem

### 2.1 Formulae for Testing Word Similarity

Our empirical formulae to be constructed test some hypothesis about word similarity. We consider only the languages where the word's base (the morphologically invariant part) is located at the beginning of the word (and not, say, at the end). It is generally true for the European languages.

Thus the formula relies on the following characteristics of the pair of words:

*n*: the total number of *final* letters differing in the two words,
*y*: the length of the common *initial* substring,
*s*: the total number of letters in the two words,

so that $2y + n = s$. For example, for the words *sadly* and *sadness*, the maximal common initial substring is *sad-*, thus the differing final parts are *-ly* and *-ness* so that $n = 6$, $y = 3$, and $s = 12$.

We consider each formula as a model from a given class The models differ in the number of parameters, which define the model complexity. Our problem is to select the optimal complexity of the model.

In this paper we will consider the following class of models for making decisions about word similarity: Two words are similar if and only if the relative number of their differing final letters is less than some threshold depending on the number of initial coincident letters of the two words:

$$\frac{n}{s} \le F(y), \qquad F(y) = a + b_1 y + b_2 y^2 + \ldots + b_k y^k, \tag{1}$$

where $n$, $s$, and $y$ are as defined above, $F(y)$ is the model function. Such function presentation is general enough because any continuous function can be represented as a convergent polynomial series.

Obviously, such models have two degrees of liberty, $n$ and $y$, with respect to the characteristics of the word pair One can also consider a model in the form $n/s \le F(y/s)$ with only one degree of liberty since $y/s = (1- n/s)/2$; however, the form (1) is more flexible, as our experiments clearly prove.

Note that more general classes of models can be suggested, as, for example, the following one: Two words are similar if the distance between them satisfies the inequality

$$D(y,s_1,s_2) \le 0, \qquad D(y,s_1,s_2) = \Sigma F(t)*\delta(t),\ t = (y+1), \ldots, N \tag{2}$$

where $s_1$ and $s_2$ are the lengths of the words, N is accepted maximum number of letters in words of a given language, $(t)$ is an indicator, $F(t)$ is a model (penalty) function. Here $(t)=\{0, 0.5, 1\}$ if there are zero, one or two letters in a correspondent position $t$ of both words, and $F(t)$ can be represented in a polynomial form, inverse polynomial form, or in any other form. However, in this paper we concentrate on the models in the form (1).

Our first task is to find the best form of model function, i.e., for the models (1), the degree of the polynomial. Then, after the model has been selected, it is easy to determine its optimal parameters.

## 2.2 Limitations of the Approach

First, our approach for testing word similarity is not applicable to irregular words. Indeed, it is impossible to construct a simple formula that could detect any similarity between English irregular verbs such as *buy* and *bought*, because these words have only one common letter in the initial part of the string. The same situation occurs in the Romance languages considered in this paper.

Since the empirical formula is constructed on the basis of statistical regularities of a language, it leads to the above-mentioned errors of the first and the second kinds (false positive and false negative). Tuning the model parameters we can control the balance between these two kinds of errors, but not to completely avoid the errors.

Some specific errors can be caused by fuzzy sense of some base meanings; we call such kind of errors the errors of 3-rd type. As an example, consider the words *move*, *moving*, *moveable*, and *moveability*. The latter two words can be considered either as

having the same basic meaning as the former two or as differing from them in the additional meaning 'ability'. Any extended interpretation of base meanings by the user leads to constructing a formula with a high level of errors of the first kind. In our example it is better to consider the similarity between the words *move* and *moving* reflecting the base meaning *movement*, and between *moveable* and *moveability* reflecting the basic meaning *ability to movement*. In this case, the formula has significantly lower error level. Obviously the errors of the 3-rd kind are more typical for document sets in special domains [Porter, 1980] For example, the words *relate* and *relativity* can be considered similar in general texts but not in texts on physics.

## 3 Inductive Method of Model Self-Organization

### 3.1 Contents of the Method

The inductive method of model self-organization (IMMSO) [Ivahnenko, 1980] allows determining a model of optimal complexity for various processes or phenomena The method chooses a near-optimal model in a given class of models using experimental data. It cannot find the very optimal model in a continuous class because it is based on competition of models; this is why the method is called *inductive*

The method consists in the following steps:

(1) An expert defines a sequence of models, from simplest to more complex ones.
(2) Experimental data are divided into two data sets: training data and test data, either manually or using an automatic procedure.
(3) For a given kind of model, the best parameters are determines using the training data and then using the test data Here any internal criteria[1] of concordance between the model and the data may be used, e.g., the least squares criterion.
(4) Both models are compared on the basis of external criteria (see below), such as the criterions of regularity and unbiasedness.
(5) The external criteria (or the most important one) are checked on having reached a stable optimum. In this case the search is finished. Otherwise, more complex model is considered and the process is repeated from the step 3.

Here is why the external criteria reach an optimum (minimum). The experimental data is supposed to contain: (a) a regular component defined by the model structure and (b) a random component—noise. Obviously, the model is to be capable to reflect the changes of the regular component. When the model is too simple, it is weakly sensible to this regular component and insensible to the noise. When the model is too complex, it reflects well the regular component but also the changing of the random component. In both cases the values of the penalty function (criterion) are large. So, we expect to find a point where the criterion reaches its minimum The principle of model *auto-organization* consists in that an external criterion passes its minimum when the complexity of the model is gradually increased

[1] Internal criteria use the same data for both evaluation of model quality and defining its parameters External criteria use different data for these purposes. Usually the external criterioa are constructed as non-negative functions with the best value 0

### 3.2 Application of the Method

In order to apply IMMSO to the problem of construction of the formulae, we consider the extreme cases of the equation (1), i e., examples of word pairs for which (1) becomes an equality. This gives a system of linear equations with respect to unknown parameters $a, b_1, b_2, \ldots, b_k$:

$$n_i/s_i = a + b_1 y_i + b_2 y_i^2 + \ldots + b_k y_i^k, \quad i = 1, \ldots, m \tag{3}$$

Here $n$, $s$, and $y$ are defined as in Section 2 1, and $i$ is the number of examples prepared by an expert on the given language. Of course, the number of equations $m$ should be more than the number of variables $(k + 1)$. To filter out the noise, the number of equations should be at least 3 times greater than the number of variables.

The examples forming the system (3) must be prepared by an expert on the given language and can be considered experimental data. Consider, for example, the words *hoping* and *hopefully*. These two words have very short common part *hop-* and long different parts *-ing* and *-efully*. The corresponding equations are:

$$9/15 = a + 3b_1 \quad \text{for linear model,}$$
$$9/15 = a + 3b_1 + 9b_2 \quad \text{for quadratic model, etc.}$$

The next steps according to IMMSO methodology are: for several $k$ starting with $k = 1$, the best solution of (4) for a given internal criterion is found and then the external criterion(s) are checked on having reached the minimum.

### 3.3 External Criteria

Generally IMMSO uses the following two criteria:

- criterion of regularity
- criterion of unbiasedness

Both criteria use the training data set and the test data set. The criterion of regularity reflects the difference between the model and the testing data, while the model is constructed on the training data set. So, this criterion evaluates the stability of the model with respect to data variation. The criterion of unbiasedness reflects the difference between the two models—those constructed on the training and on the testing set, respectively. So, this criterion evaluates independence of the model from the data.

Different forms of these criteria can be proposed, a specific form depending on the problem. In our case we use these criteria in the following forms:

$$K_r = \frac{\sqrt{\sum_C \left(q_i(T) - q_i\right)^2}}{\sqrt{\sum_C \left(q_i\right)^2}} \qquad K_u = \frac{\sqrt{\sum_{T+C} \left(q_i(T) - q_i(C)\right)^2}}{\sqrt{\sum_{T+C} \left(q_i\right)^2}} \tag{4}$$

Here $T$ and $C$ are the systems of equations (3) used for training and testing, respectively; $q_i(T)$ and $q_i(C)$ are the "model" data that is the right part of equations with the parameters determined on the data of training and testing, respectively; $q_i$ are the "experimental" data, i.e., the left part of the equations; $i$ is the number of the equation.

Sometimes a model can be better than another one according to the first criterion but worse according to the second one. Then a combined criterion is used, e.g.:

$$K = \lambda K_r + (1-\lambda) K_u, \tag{5}$$

where $\lambda$ is a user-defined coefficient of preference. In our experiments with Romance languages we use $\lambda = 2/3$, i.e., we consider the criterion of regularity as the main one.

## 4 Constructing Empirical Formula for Romance Languages

### 4.1 Selection of Examples

The formula to be constructed is considered as the first approximation and may be later tuned on the texts from a given domain. So, our examples selected for training and control reflect only some general regularities of a language.

We assumed that: (a) the initial common parts of similar words were distributed uniformly between the shortest and the longest ones; (b) the final parts of similar words were also distributed uniformly between the shortest and the longest ones. So, we took 50% of word pairs with the shortest common part and 50% with the longest common part. In each pair, we tried to take the words with the short and long final parts. We did not consider the words containing less than 4 letters and we removed diacritics from letters (i.e., ê, é, è → e, ç → c).

The number of examples taken for training and testing was 10, which corresponded to the expected maximum number of model parameters of 4 (cubic polynomial).

### 4.2 Training Procedure

For training procedure, we considered the following pairs of words having the same basic meanings:

**French**

| N | Short words | | N | Long words | |
|---|---|---|---|---|---|
| 1. | Blanc | Blanc*heur* | 6. | Impossib*le* | Impossib*ilite* |
| 2. | Pleur*ant* | Pleur*erait* | 7. | Degenere*r* | Degenere*scent* |
| 3. | Mange*ur* | Mange*rent* | 8. | Macadam | Macadam*iser* |
| 4. | Guet | Guet*teurs* | 9. | Pauvre*te* | Pauvre*ment* |
| 5. | Bless*ant* | Bless*ures* | 10. | Abrutiss*ant* | Abrutiss*ement* |

**Italian**

| N | Short words | | N | Long words | |
|---|---|---|---|---|---|
| 1. | Sebo | Seborrea | 6. | Convertire | Convertibile |
| 2. | Bello | Belta | 7. | Convinzione | Convincimento |
| 3 | Arte | Artistico | 8 | Fiammifero | Fiammeggiare |
| 4. | Casa | Casigliano | 9. | Macchinatore | Macchinazione |
| 5. | Alter | Alterarsi | 10. | Panellenico | Panellenismo |

**Portuguese**

| N | Short words | | N | Long words | |
|---|---|---|---|---|---|
| 1. | Dossel | Dosseladas | 6 | Abandonando | Abandonamento |
| 2. | Dotar | Dotacaos | 7. | Amoravel | Amoravelmente |
| 3 | Hipnose | Hipnotico | 8. | Celebridades | Celebrizaram |
| 4. | Janta | Jantarada | 9. | Catalogacao | Catalogadora |
| 5. | Jardim | Jardineiro | 10 | Caracteri-zante | Caracterizador |

**Spanish**

| N | Short words | | N | Long words | |
|---|---|---|---|---|---|
| 1. | Celo | Celosamente | 6 | Arrogante | Arrogancia |
| 2. | Cazar | Cazador | 7 | Institucional | Institucionalmente |
| 3. | Arte | Artistico | 8. | Multiplicados | Multiplicaciones |
| 4. | Comer | Comida | 9. | Descentralizados | Descentralizables |
| 5. | Altura | Altitud | 10. | Caracteristica | Caracterizaremos |

With these examples, we obtained the following systems of linear equations for French:

| N | $0^{th}$ order model |
|---|---|
| 1 | $4/14 = a$ |
| 2. | $8/18 = a$ |
| 3. | $6/16 = a$ |
| 4 | $5/13 = a$ |
| 5 | $7/17 = a$ |
| 6. | $7/23 = a$ |
| 7. | $6/22 = a$ |
| 8. | $4/18 = a$ |
| 9. | $6/18 = a$ |
| 10. | $8/24 = a$ |

| N | Linear model |
|---|---|
| 1. | $4/14 = a + 5b_1$ |
| 2. | $8/18 = a + 5b_1$ |
| 3 | $6/16 = a + 5b_1$ |
| 4. | $5/13 = a + 4b_1$ |
| 5. | $7/17 = a + 5b_1$ |
| 6. | $7/23 = a + 8b_1$ |
| 7 | $6/22 = a + 8b_1$ |
| 8. | $4/18 = a + 7b_1$ |
| 9. | $6/18 = a + 6b_1$ |
| 10. | $8/24 = a + 8b_1$ |

| N | Quadratic model |
|---|---|
| 1. | $4/14 = a + 5b_1 + 25b_2$ |
| 2 | $8/18 = a + 5b_1 + 25b_2$ |
| 3. | $6/16 = a + 5b_1 + 25b_2$ |
| 4. | $5/13 = a + 4b_1 + 16b_2$ |
| 5. | $7/17 = a + 5b_1 + 25b_2$ |
| 6 | $7/23 = a + 8b_1 + 64b_2$ |
| 7. | $6/22 = a + 8b_1 + 64b_2$ |
| 8. | $4/18 = a + 7b_1 + 49b_2$ |
| 9. | $6/18 = a + 6b_1 + 36b_2$ |
| 10. | $8/24 = a + 8b_1 + 64b_2$ |

Similar linear system was also constructed for the cubic model. Using the least squares method, we found four sets of model parameters ($a$, $b_1$, ...) for each of the mentioned models, to be further used in the external criterions

The models for Italian, Portuguese, and Spanish were constructed in a similar way

### 4.3 Testing Procedure

For testing procedure, we considered the following pairs of words having the same base meanings:

**French**

| N | Short words | | N | Long words |
|---|---|---|---|---|
| 1. | Froid | Froid*eur* | 6. | Emine*mment* |
| 2. | Mort | Mort*elle* | 7. | Epigramme | Epigramm*atique* |
| 3. | Fer*ais* | Fe*rrer* | 8. | Retentiss*ant* | Retentiss*ement* |
| 4. | Fin | Fin*alite* | 9. | Constitutionn*el* | Constitutionn*alisme* |
| 5. | Lutte*r* | Lutt*ait* | 10. | Difficulte | Difficult*ueux* |

**Italian**

| | Short words | | | Long words | |
|---|---|---|---|---|---|
| 1. | Certo | Cert*ezza* | 6. | Perfora*tora* | Perfora*zione* |
| 2. | Forc*a* | For*chetta* | 7. | Periodi*co* | Periodi*zzare* |
| 3. | Mostr*o* | Mostr*uosita* | 8. | Cattedra | Cattedr*atico* |
| 4. | Pane | Pan*ettone* | 9. | Cattechi*smo* | Cattechi*zzatore* |
| 5. | Balle | Balle*rina* | 10. | Necessaria | Necessari*amente* |

**Portuguese**

| | Short words | | | Long words | |
|---|---|---|---|---|---|
| 1. | Pequen*eza* | Pequen*inas* | 6. | Descompassa*r* | Descompassa*damente* |
| 2. | Doura*r* | Doura*dura* | 7. | Experimento*u* | Experimen*tavel* |
| 3. | Macho | Mach*ista* | 8. | Impugna*r* | Impugna*tivo* |
| 4. | Nodo | Nodo*sidade* | 9. | Doutora*l* | Doutora*mento* |
| 5. | Nome | Nom*inalidade* | 10. | Necessi*tadas* | Necessi*dades* |

**Spanish**

| | Short words | | | Long words | |
|---|---|---|---|---|---|
| 1. | Circo | Circ*ense* | 6. | Sentimentale*s* | Sentimentali*smo* |
| 2. | Afan | Afan*oso* | 7. | Discrimina*mos* | Discrimina*cion* |
| 3 | Denso | Dens*idad* | 8. | Legisla*dor* | Legisla*tura* |
| 4. | Cabe*r* | Cab*ida* | 9 | Especialis*ta* | Especializ*arse* |
| 5. | Creado | Creac*ion* | 10. | Necesario | Necesari*amente* |

On the basis of these examples we constructed the correspondent systems of linear equations for the four models like we did in the training procedure. Using the least squares method, we found again the four sets of the model parameters ($a$, $b_1$, ....) for each of the mentioned models. Of course, these models differed from the models constructed on the testing set.

### 4.4 Results

Using the criteria (4) and (5), we found that the linear model proved to be the winner for all languages. The results are summarized in the following tables:

**French**

| | 0-order model | Linear model | Quadratic model | Cubic model |
|---|---|---|---|---|
| Criterion $K_r$ | 0.28 | **0.20** | 0.29 | 5.27 |
| Criterion $K_u$ | 0.01 | **0.04** | 0.18 | 3.81 |
| Criterion $K$ | 0.19 | **0.15** | 0.25 | 4.78 |

**Italian**

| | 0-order model | Linear model | Quadratic model | Cubic model |
|---|---|---|---|---|
| Criterion $K_r$ | 0.28 | **0.20** | 0.21 | 0.33 |
| Criterion $K_u$ | 0.15 | **0.10** | 0.14 | 0.20 |
| Criterion $K$ | 0.24 | **0.17** | 0.19 | 0.29 |

**Portuguese**

| | 0-order model | Linear model | Quadratic model | Cubic model |
|---|---|---|---|---|
| Criterion $K_r$ | 0.31 | **0.23** | 0.25 | 2.25 |
| Criterion $K_u$ | 0.13 | **0.14** | 0.17 | 2.41 |
| Criterion $K$ | 0.25 | **0.20** | 0.22 | 2.30 |

**Spanish**

| | 0-order model | Linear model | Quadratic model | Cubic model |
|---|---|---|---|---|
| Criterion $K_r$ | 0.26 | **0.19** | 0.18 | 0.23 |
| Criterion $K_u$ | 0.09 | **0.11** | 0.12 | 0.18 |
| Criterion $K$ | 0.20 | **0.16** | 0.16 | 0.21 |

At the last step for all languages we considered only the linear model. We joined together the training set and the testing set of examples and obtained the linear system of 20 equations and 2 variables. The solutions gave the following formulae for testing word similarity:

French: $n/s \leq 0.481 - 0.024\,y$
Italian: $n/s \leq 0.571 - 0.035\,y$
Portuguese: $n/s \leq 0.528 - 0.029\,y$
Spanish: $n/s \leq 0.549 - 0.029\,y$

Similarly, joining together all 80 examples we determined the generalized formula:

$$n / s \leq 0.530 - 0.029\, y$$

This formula can be considered an initial approximation for further tuning on other romance languages.

## 5 Experimental Results

### 5.1 Document Collections

The constructed formulae were checked on real document collections. The goals of these experiments were:

- To compare the quality of the formulae on different languages,
- To reveal the sensibility of each formula to its parameters

We considered the documents on the popular theme: mortgage and crediting. This theme is narrow enough to provide a representative set of similar words. We took 6 articles in each language containing in total from 16000 to 24000 words (excluding numbers and words with less then 4 letters). The statistical characteristics of style for each document set were rather close, namely: 22.6–24.4 for text complexity and 0.14–0.17 for word's resource variety. The first figure is calculated with as $n * \ln m$ and the second one as $N / M$, where $n$ and $m$ are the average length of words (in letters) and phrases (in words), $N$ and $M$ are the number of different words (before grouping) and that of all words. Therefore, the conditions for all languages were the same.

To reduce the number of comparisons, we randomly selected some paragraphs from the mentioned document collections. Since we assumed that similar words had different final parts, it was natural to order all words alphabetically and to compare the neighbors (this is not necessarily the best way for grouping similar words, but we used it to simplify manual testing of word pairs). As a result we obtained the alphabetical lists of words partially presented in the following table:

| French | Italian | Portuguese | Spanish |
|---|---|---|---|
| absence | accedere | abrange | abaratado |
| accepte | acquistare | abrangencia | abogado |
| accord | acquisto | abrangendo | abogados |
| accorde | adatte | acessoes | acceder |
| accordee | adatto | acima | actual |
| ...etc... | ...etc... | ...etc... | ...etc... |
| In total: 456 words | In total: 567 words | In total: 506 words | In total: 536 words |

### 5.2 Results

In our experiments we tested similarity of adjacent words automatically and manually with different combinations of parameters used in the formulae, varied in ±10% for each parameter. The quality of the results was estimated by the total statistical error

$P_{err} = P_p + P_n$ and F-measure of accuracy $F = 2 / (1/R + 1/P)$. Here $P_p$ and $P_n$ are the probabilities of the statistical errors of the 1-st and the 2-nd kind; R and P are recall and precision. The first estimation is usually used in mathematical statistics [Cramer, 1946], while the second one in information retrieval [Baeza-Yates, 1999].

The following tables show the result of automatic processing for different languages.

**French** (455 tests = 115 similar + 340 non-similar)

| Parameters | 0.48, -0.024 | 0.43, -0.024 | 0.53, -0.024 | 0.48, -0.021 | 0.48, -0.027 |
|---|---|---|---|---|---|
| Similar cases | 104 | 88 | 126 | 111 | 100 |
| Not similar | 351 | 367 | 329 | 344 | 355 |
| False alarm | 7 | 3 | 18 | 10 | 7 |
| Omission | 18 | 30 | 7 | 14 | 22 |
| False positive $P_p$ | 2.0% | 0.9% | **5.3%** | 2.9% | 2.0% |
| False negative $P_n$ | 15.7% | 26.1% | **6.1%** | 12.2% | 19.1 |
| Recall R | 84.3% | 73.9% | **93.9%** | 87.8% | 80.9% |
| Precision P | 93.3% | 96.6% | **85.7%** | 91.0% | 93.0% |
| Summary | Min $P_{err}$ = 11.4% | | Max F = 89.7% | | |

**Italian** (566 tests = 140 similar + 426 non-similar)

| Parameters | 0.57, -0.035 | 0.51, -0.035 | 0.63, -0.035 | 0.57, -0.031 | 0.57, -0.039 |
|---|---|---|---|---|---|
| Similar cases | 149 | 120 | 193 | 166 | 126 |
| Not similar | 417 | 446 | 373 | 400 | 440 |
| False alarm | 39 | 19 | 65 | 45 | 19 |
| Omission | 30 | 39 | 12 | 19 | 33 |
| False positive $P_p$ | 9.2% | 4.2% | **15.3%** | 10.6% | 4.5% |
| False negative $P_n$ | 21.4% | 27.9% | **8.6%** | 13.6% | 23.6% |
| Recall R | 78.6% | 72.1% | 91.4% | 86.4% | **76.4%** |
| Precision P | 73.8% | 84.2% | 66.8% | 72.9% | **84.9%** |
| Summary | Min $P_{err}$ = 23.9% | | Max F = 80.3% | | |

**Portuguese** (505 tests = 138 similar + 367 non-similar)

| Parameters | 0.53, -0.029 | 0.48, -0.029 | 0.58, -0.029 | 0.53, -0.026 | 0.53, -0.032 |
|---|---|---|---|---|---|
| Similar cases | 136 | 117 | 159 | 141 | 132 |
| Not similar | 369 | 388 | 346 | 364 | 373 |
| False alarm | 15 | 9 | 27 | 17 | 14 |
| Omission | 17 | 30 | 6 | 14 | 20 |
| False positive $P_p$ | 4.1% | 2.5% | **7.4%** | 4.6% | 3.8% |
| False negative $P_n$ | 12.3% | 21.7% | **4.3%** | 10.1% | 14.5% |
| Recall R | 87.7% | 78.3% | **95.7%** | 89.9% | 85.5% |
| Precision P | 89.0% | 92.3% | **83.0%** | 87.9% | 89.4% |
| Summary | Min $P_{err}$ = 11.7% | | Max F = 89.3% | | |

**Spanish** (535 tests = 165 similar + 370 non-similar)

| Parameters | 0.55, -0.029 | 0.49, -0.029 | 0.61, -0.029 | 0.53, -0.026 | 0.53, -0.032 |
|---|---|---|---|---|---|
| Similar cases | 164 | 128 | 183 | 167 | 142 |
| Not similar | 371 | 407 | 352 | 368 | 393 |
| False alarm | 22 | 8 | 34 | 23 | 14 |
| Omission | 23 | 45 | 16 | 21 | 37 |
| False positive $P_p$ | 5 9% | 2 2% | **9.2%** | 6.2% | 3 8% |
| False negative $P_n$ | 13.9% | 27.3% | **9.7%** | 12.7% | 22.4% |
| Recall R | 86.1% | 72 7% | 90 3% | **87.3%** | 77 6% |
| Precision P | 86.6% | 93.8% | 81.4% | **86.2%** | 90.1% |
| Summary | Min $P_{err}$ = 18.9% | | Max F = 86.6% | | |

Examples of errors of all kinds are presented at the following table:

| Errors | **French** | **Italian** | **Portuguese** | **Spanish** |
|---|---|---|---|---|
| 1st kind | commune<br>communique | casa<br>caso | entre<br>entrega | ahora<br>ahorro |
| 2nd kind | hypotheque<br>hypothecaire | iniziali<br>inizialmente | entendemos<br>entendimiento | invertido<br>inversores |
| 3rd kind | simple<br>simplifie | numero<br>numerose | titulo<br>titulares | bancarios<br>bancarrota |

## 6 Conclusions

We have suggested a knowledge-poor approach for testing word similarity. Our empirical formulae do not require any morphological dictionaries of the given language and can be constructed manually or automatically basing on few examples. This is useful for constructing word frequency lists when working with multilingual databases and multi-thematic document collections.

Our experiments with Romance languages show that our approach provides the 80%–90% accuracy (F-measure), committing 2%-5% of the errors of the 1-st kind and 20%-25% of the 2-nd kind. This is rather acceptable in semi-automatic setting since the human expert can easily join the similar words after the grouping procedure.

In the future we plan to construct several other empirical formulae and compare them with those reported in this paper. We plan to give linguistic explanation for the behaviour of constructed formulae if possible. We plan also to take into account some statistical regularities extracted from the training document set. We thank A. Gelbukh and M. Porter for useful suggestions on such modifications.

## References

1 Baeza-Yates, R., Ribero-Neto, B. (1999): *Modern Information Retrieval*. Addison Wesley.

2 Cramer, H. (1946): *Mathematical methods of statistics*. Cambridge

3 Gelbukh, A. (2003): Exact and approximate prefis search under access locality requirements for morphological analysis and spelling correction. *Computación y Sistemas*, vol. 6, N 3, 2003, pp. 167–182.

4 Gelbukh, A., G. Sidorov (2001): Zipf and Heaps Laws' Coefficients Depend on Language. *Computational Linguistics and Intelligent Text Processing* (CICLing-2001) Lecture Notes in Computer Science, N 2004, Springer, pp 332–335.

5 Gelbukh, A , G Sidorov (2002): Morphological Analysis of Inflective Languages through Generation. *Procesamiento de Lenguaje Natural*, No 29, 2002, p. 105–112.

6 Gelbukh, A , G Sidorov (2003): Approach to construction of automatic morphological analysis systems for inflective languages with little effort. In: *Computational Linguistics and Intelligent Text Processing* (CICLing-2003), Lecture Notes in Computer Science N 2588, Springer, pp. 215–220.

7 Ivahnenko, A (1980): *Manual on typical algorithms of modeling* Tehnika Publ., Kiev (in Russian).

8 Makagonov, P., M. Alexandrov (2002): *Constructing empirical formulas for testing word similarity by the inductive method of model self-organization*. In: Ranchhold and Mamede (Eds.) "Advances in Natural Language Processing", Springer, LNAI, N 2379, pp. 239–247.

9 Porter, M. (1980): An algorithm for suffix stripping. *Program*, 14, pp 130–137.