Induction Trees for

Automatic Word Classification

Daniela López De Luise
AI Group , Facultad de Ingeniería, Universidad de Palermo (UP)
Ciudad Autónoma de Buenos Aires – Argentina
lopezdeluise@yahoo.com.ar

and

Juan M. Ale
Facultad de Ingeniería, Universidad de Buenos Aires(UBA)
Ciudad Autónoma de Buenos Aires – Argentina
ale@acm.org.ar

Abstract

This work studies induction tree application for certain word category detection by simple morpho-syntactical descriptors that are proposed here. The classification power for these new descriptors with and without stemming is also studied. Finally, results show that classification prediction power is good when stem is coordinated with a short list of descriptors.

Keywords: machine learning, lexical categorization, morphology, syntax

Resumen

En este trabajo estudia el uso de árboles de inducción para la detección de ciertos tipos de palabras usando algunos descriptores morfosintáctico propuestos. También se estudia el poder de clasificación de estos nuevos descriptores con y sin extracción de raíces de palabras (stemming). Finalmente, se muestra en los resultados que el poder de predicción de la clasificación es bueno cuando se combinan stemming con algunos de los descriptores presentados.

Palabras claves: aprendizaje automático, clasificación de palabras, mofología, sintaxis

1. INTRODUCTION

It is hard to perform an efficient handling of digital documentation due to several phenomena as synonymy (different words with similar meaning), polysemy (a word with two or more meanings), anaphoras (implicit mentions by means of demonstrative pronouns), metaphors (use of a word with a meaning or in a context different from the habitual one), metonymy (rhetorical figure that consists of transferring the meaning of a word or phrase to another word or phrase with different meaning, with semantic or logical proximity) [10], misspellings, punctuation, neologisms, foreigner words and differences between linguistic competence (based in grammar rules) and actuation (the way grammar is used by a native speaker) [2]. Many approaches have been used to solve these problems, some of them are:
-Exhaustive tables of words or punctuation, optionally combined with lexical knowledge databases such as WordNet (to process using synonyms) [10].
-Exhaustive text revision to extract and classify errors in texts [2].
-Use of a corpus of traditionally detectable mistakes in the language [2].
-Normative [2].
-Style books [2].
-Scoring synonymy degree of expressions [4].
-Contextual information processing [13].

Based on those strategies, several applications and studies have been performed: for correcting documents [14], classification of documents, written text analysis, inflectional language[1] analysis [17], statistical machine translation [12], text summarization [10], automatic grammar and style checking [2] automatic translation [4], etc., even covering areas like statistical modeling of speech [8]. To perform such activities it is very useful to be able to automatically detect the word lexical category (if a word is a noun, article, verb, etc.). Sometimes this detection is part of the global approach as in the case of the text checking presented in [6], whereas in other cases are special developments as in [2], or [7], but always with complex semantic management or with long linguistic inference procedures. This paper proposes a set of morpho-syntactical descriptors for words, using just local information, to be used to automatically find out the actual lexical category of certain words with reasonable precision. The set of morpho-syntactical descriptors defined here are combined with stemming algorithmic [15] to get invariant radixes as extra descriptors. This proposal uses also an Induction Tree. Although Induction Trees[2] can be used for learning in many areas [11], they are applied here to word classification. An induction tree is a model of some basic characteristics of a dataset extracted by an induction process on instances. It is used due to its flexibility and its power to apply the acquired knowledge to new concrete instances.

Because the Web is a kind of text repository, traditional morpho-syntactical processing had to overcome new problems (specific problems for internet documentation): It will be required to adapt processing to activities such as Web Services [14], Information Retrieval, automatic extraction of knowledge from Web Documents [1], using Web as a

---

[1] languages, where words have usually several different morphological forms that are created by changing a suffix [17].

[2] From Mitchell [11]: “Decision Tree learning is a method for approximating discrete-valued target functions, in which the learned function is represented by a decision tree. Learning trees can also be re-represented as sets of if-then rules to improve human readability. These learning methods are among the most popular of inductive inference algorithms and have been successfully applied to a broad range of tasks from learning to diagnose medical cases to learning to assess credit risk on loan applicants” .

corpus for automatic collocation[3] identification [16], etc. Therefore it is important to process automatically text as mentioned previously but considering the special features of web writers and readers. For that reason, all the text processed in this paper is extracted only from web pages.

Another point is that internet sets the same availability degree for sites in any language. So, the web pages covered here are taken from Spanish sites in any country.

The rest of this paper is organized as follows: section 2 describes the database and data collection procedure, section 3 describe field selection and induction tree model construction, and section 4 presents some conclusions and future work.

2. DATA ANALYSIS

In this section there is a short description of the processing steps (section 2.1), dataset and sample characteristics (sections 2.2 and 2.3 respectively).

2.1. Methodology

Four sets of web pages in Spanish were made regarding several topics. All of them were downloaded in text format. From the total number of 340 pages, 361217 words were extracted with a Java application. The output was saved as 15 plain text files. The text files were converted into Excel format to be able to use an Excel's form to manually fill in the field tipoPalabra (kind of word). The resulting files were processed with other java program to introduce the stemming column and afterward converted into csv format to be able to work with WEKA[4] software. After that, some preliminary statistics were performed with InfoStat[5] to detect the main dataset features and the csv files were processed with WEKA Explorer. An induction tree model was built from data as detailed in the following sections. Figure 1 depicts graphically all the mentioned steps.

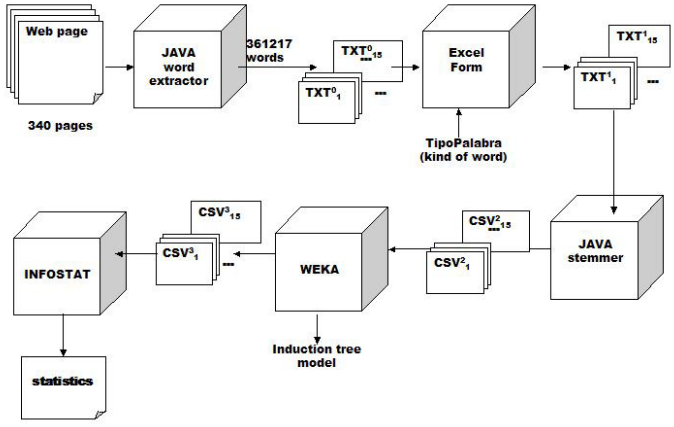


Figure 1. Flow of processing steps

[3] statistically significant word associations representing "a conventional way of saying things" [9].
[4] WEKA: open source workbench for Data Mining and Machine Learning [18].
[5] InfoStat: statistical software from a group named InfoStat in the Universidad Nacional de Córdoba.

2.2. Dataset Description

The text files were processed with a Java application. For each word, a set of 25 description fields were extracted. Therefore, each database record represent a word. The fields are detailed below:

-Continue fields: there isn't.

-Numerable fields: 10 fields were non-negative integers with a big boundary (see Table 1). All of them were discretized into fixed-size intervals, to be able to categorize and process them together with nominal fields. They were separated into 3 or 5 categories. (see Table 2).

-Discrete fields: there isn't.

-No-numeric fields: 15 fields have a domain composed by a specific set of literals (syllabus, punctuation signs, a set of predefined words or the classical binomial Yes/No). See Table 3 for details.

-Missing data: they were considered as a distinct data value and processed with the rest of the data.

Table 1. Numerable fields

Field	description
Id-caso	Web page dentifier
Web-profundidad-pagina	Number of slashes ("/") in the web page URL
Cant-ocurrencias	Times the word is repeated in the page
Cant-pal-pagina	Number of words in the page
Long-palabra	Number if characters in the word
Cant-vocales-fuertes	Number of strong vowels in the word (a, e, o)
Cant-vocales-debiles	Number ofweak vowels in the word (I, u)
Long-oracion	Number of words (in words' sentence)
Cant-numeros	Quantity of numbers (in words' sentence)
Cant-signos-especiales	Number of special characters (in words' sentence)

Table 2. Categorization

Field	categories	Max value
Web-profundidad-pagina	5	7000
Cant-ocurrencias	3	1168000
Cant-pal-pagina	5	6792600
Long-palabra	5	31000
Cant-vocales-fuertes	3	11000
Cant-vocales-debiles	3	6000
Long-oracion	5	259000
Cant-numeros	5	842000
Cant-signos-especiales	5	149000

2.3 Sample Characteristics

Data fields dependences were studied with correspondence analysis. This task was performed with InfoStat software. All the 25 fields were considered, but only a random sample of 47820 instances were processed. The independency test was performed with parameter $\alpha = 0.05$, statistic χ^2 y H_0= "independent".

Results show that:

-tipoPalabra (kind of word) is independent from tipoPag (kind of page) and siguePuntuación (punctuation follows the actual word).

-palAntTipo (kind of previous word) is independent from cantVocalesFuerte (number of strong vowels in the word).

-resaltada (the word is remarked in the text) is independent from cantVocalesFuerte (number of strong vowels).

Table 3. Results with Different Splits

Field	Description
Id-palabra	Word extracted
Tema	Main topic of the web page
Tipo-pal	One of these: sustantivo, verbo, otro (noun, verb, other respectively)
Pal-anterior	Identifier of the previous word in the sentence
Pal-ant-tipo	One of these: ninguna, otro (none, other). This indicates if it is the first word in the sentence.
Tipo-pag	One of these: indice, contenido (index, plain text).
Pais-radicacion	One of the country codes in the url: us, ar, es, cu, mx, etc.
Terminacion	Suffix of the word: ar, er, ir, or, ur, ra, re, ri, ro, ru, s, m, sa, se, si, so, su, an, en, in, on, un, cion, ciones, null.
Sigue-puntuacion	One of these: si, no (Yes, No). This value is "Si" when one the following is after the word: ".", ",", ";", ":".
Clase-pag	One of these: org, com, net, otro. Indicates the kind of domain from the URL.
Empieza-mayuscula	One of these: si, no (Yes/No). This value is "Si" when the word starts capitalized.
Resaltada	One of these: si, no (Yes/No). This value is "Si" when the word is enclosed with quotation marks or is capitalized.
Es-titulo	One of these: si, no (Yes/No). This value is "Si" when the word belongs to the first sentenece in the text.
Frase-especial	One of these: si, no (Yes/No). This value is "Si" when the sentence is enclosed with: ¡!, ¿?, <>, (), [], {} or «»
Stem	Radix of the word (according to Porter algorithm)

3. INDUCTION TREES FOR CLASSIFICATION

In this section the construction of an induction tree (using WEKA [18] software) with many parameter values is studied. The remainder of this work uses the following metrics to evaluate results [18]:

1) Metrics used for error handling evaluation

1.a) Precision: metric used in Information Retrieval (IR). It is the rate of relevant instances returned by the total of instances returned.

1.b) Recall: metric used IR. It is the rate of relevant returned by the number of relevant instances.

1.c) Recall-precision: plot used in IR with recall (x-axis) and precision (y-axis).

2) Metrics used for predictability

2.d) Kappa (κ): used to compare predictability against a random predictor. It can take from 0 to 1, being 0 the random predictor value and 1 the best predictor.

3) Metrics for confidence validation

3.e) Margin curve: a bigger margin denotes a better predictor. It is the difference between the estimated probability of the true class and that of the most likely predicted class other than the true class.

In the following, an induction tree with J4.8 algorithm is used to build a model to predict the kind of certain words based on the descriptors introduced in this paper (section 3.1), based on the descriptors and stem (section 3.2) and based only in the best descriptors and stem (section 3.3).

3.1. Classification of Words Using proposed descriptors

The J48 algorithm is used here to build the induction tree using the fields presented in 2.2. Dataset Description (except for stem field). Here, the following analysis is performed: alternate splittings of training sample, different data categorizations, influence of descriptors on the model and windowing[6].

[6] windowing is a strategy for selecting a subset of data for processing.

1) Splitting of the training sample.
Different percentages of instances were taken from the same sample to construct/validate the model by setting several splitting values. The data records were randomly extracted from the 47820 instances according to the settled percentage. The initial sampling window had 6838 instances. Results are shown in Table 4.

Table 4 Results with Different Splits

split	correctly classified	Kappa statistic
66%	70.7%	0.4582
70%	70.7%	0.4558
00%	72.7%	0.4981

It can be seen that classification improves from 66% of instances for testing (and 34% for training) to 100% for training and testing. The classification model becomes more confident.

2) Alternates for field categorization.
As part of sensitivity analysis, different categorizations for just one of the descriptor variables is performed: cantOcurrencias (number of times the word is detected within the html page). This variable is selected for this study because it is always near the tree-model root (it is important to determine the kind of word). It was evaluated with 3 and 7 bins. Results are shown in Table 5.

Table 5. Results with Different Categorizations

split: 66%

categories	7	3
correctly classified	70.9%	70.7%
incorrectly classified	29.1%	29.3%
Kappa statistic	0.4698%	0.4582%

split: 70%

categories	7	3
correctly classified	70.8%	70.7%
incorrectly classified	29.2%	29.2%
Kappa statistic	0.4716%	0.4558%

split: 00%

categories	7	3
correctly classified	74.5%	70.7%
incorrectly classified	26.5%	29.3%
Kappa statistic	0.5166%	0.4981%

The table shows the precision and total error changes due to categorization. To study the strength of this tendency, the margin-curves, precision, recall and recall-precision analysis is performed but only for nouns:

- Margin-curves for 3 and 7 categories reflect a slight tendency to join the x-axis with the instance number. It seem like each new instance makes the classifier more trustable. This tendency becomes apparent with 66% of splitting, and remains with 70% and 0% (see Figure 2).

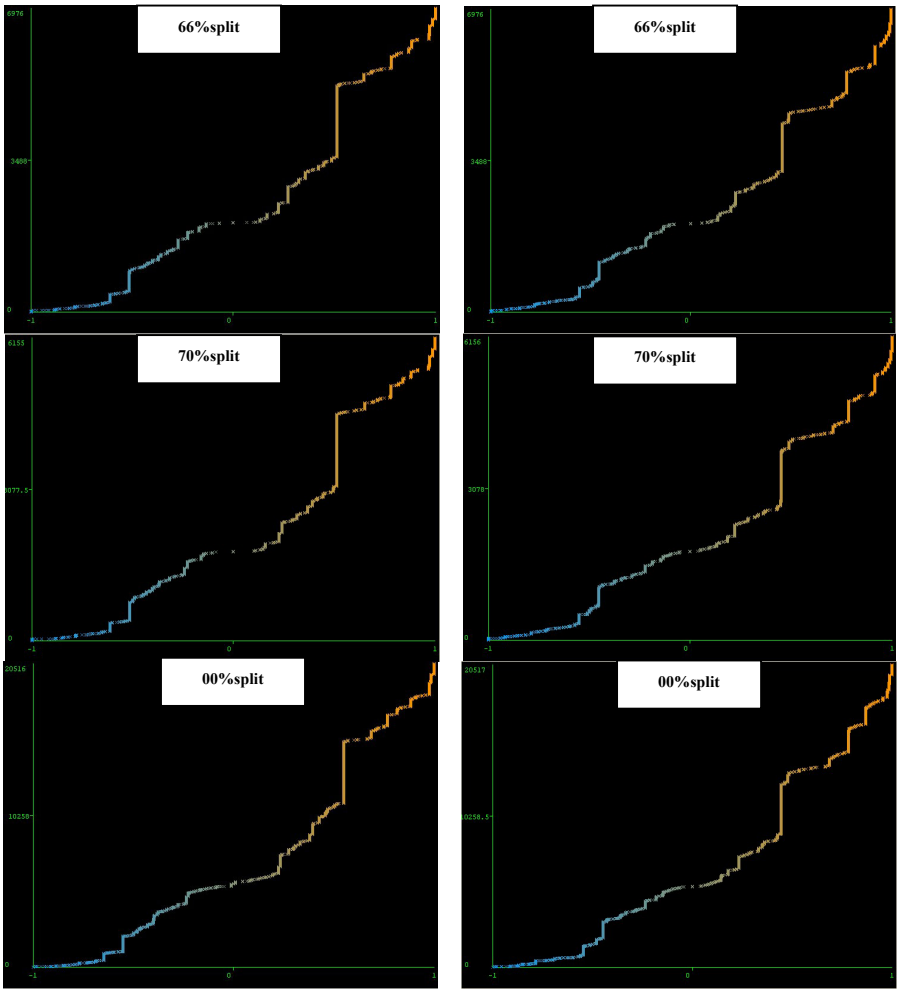
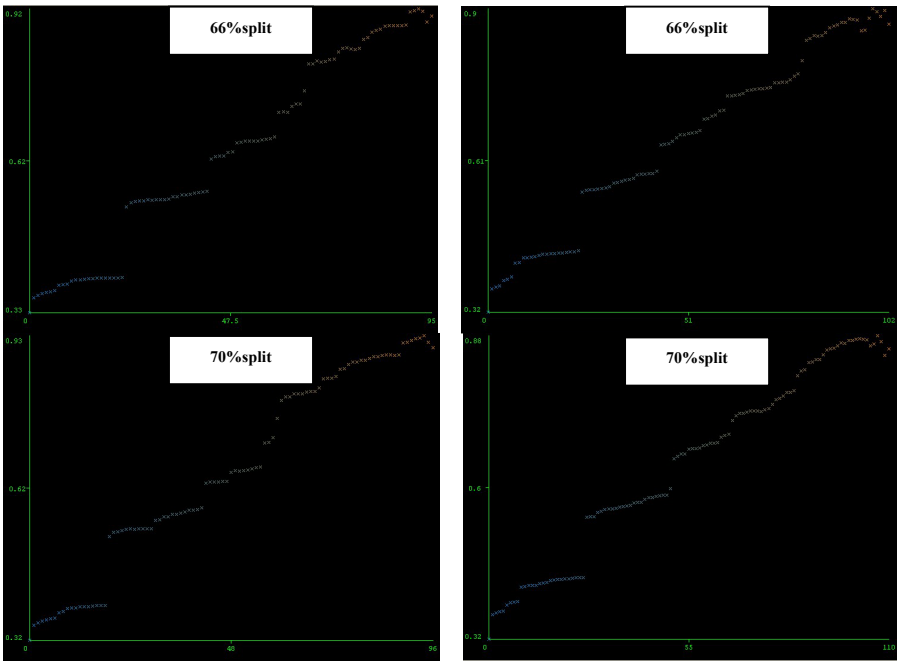


Figure 2. margin curve with 3 (on the left) and 7 categories (on the right)

- Precision-curves show that precision with 3 categories is better than with 3 categories but with 7 categories more instances are retrieved (102 against 95 with 66% of splitting). See Figure 3.



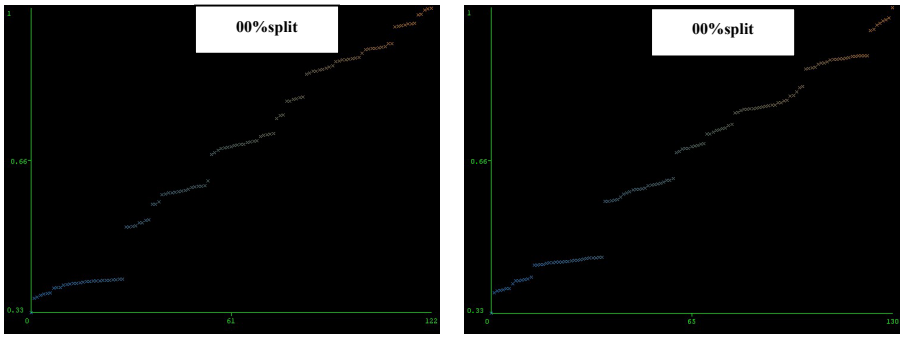


Figure 3. Precision curve for 3 (on the left) and 7 (on the right) categories

- Recall-curve presents a minimum recall value for 7 categories higher than the value for 3 categories. Conversely, the slope has a softer slope for 3 categories (see Figure 4).

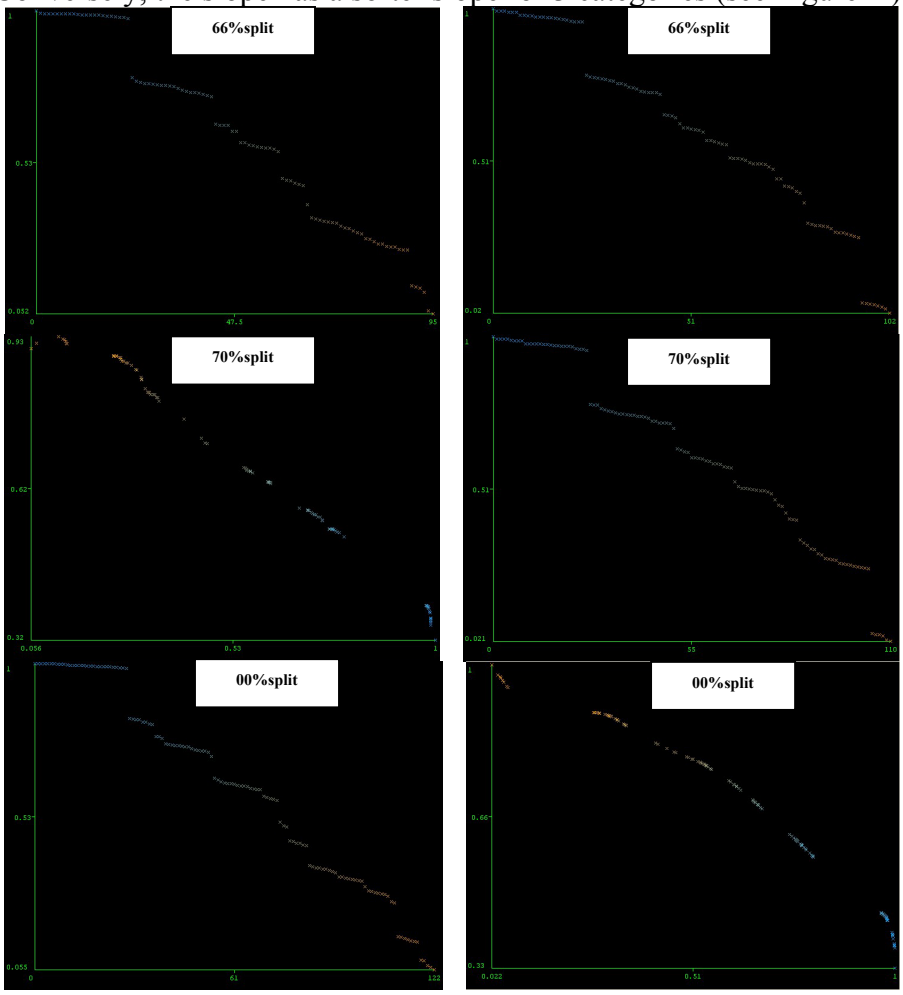


Figure 4. Recall curve for 3 (on the left) and 7 (on the right) categories

- Finally, precision-recall curve (see Figure 5), show that precision is best for 3 categories but at expense of fewer number of instances. This behavior is observed for all the splitting rates experienced (66%, 70%, 0%).

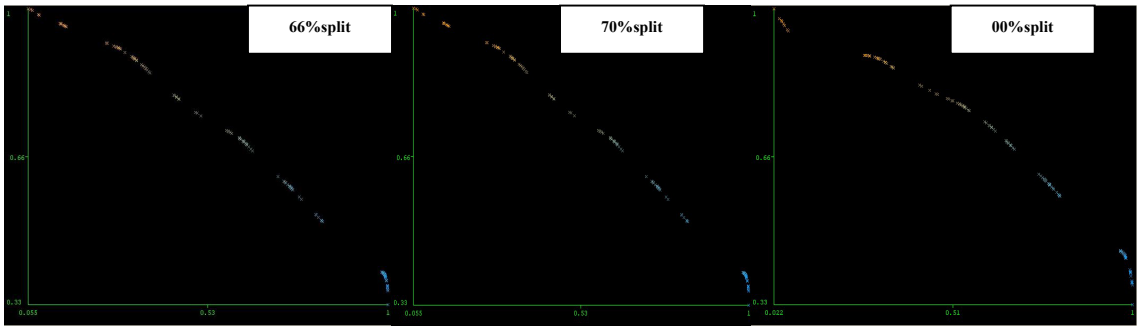


Figure 5. Precision-recall curve for 3 categories (66%, 70%, 0% split)

3)Descriptor choices

In this section alternate descriptor selection criteria are studied to find out the influence of field selection on the classification power. Table 6 shows a brief of the following analysis:

a)*Low-computational-cost fields selection*: the high-cost fields, were taken out whenever the removal did not affect the tree performance. The resulting selection was 12 descriptors. None of them involves processing all the html document. Just one of such descriptors needs sentence processing.

b)*Categorized fields selection*: nominal fields where removed and the numeric fields (categorized) were used to construct the model.

c)*Nominal fields selection*: numeric fields where removed and the nominal fields were used to construct the model.

d)*Independent fields selection*: some independent fields were taken out before constructing the model (see correspondence analysis in 2.3 Sample Characteristics). The process was repeated 5 times changing the extracted subsets according with different criteria.

Table 6. Results with Different field selections

criteria	fields	total	Classif. OK	kappa
Low cost	tema, palAntTipo, tipoPag, terminacion, empiezaMayuscula, resaltada, esTitulo, CATlongPalabra, CATcantVocalesFuerte, CATcantVocalesDebile, CATlongOracion	12	71.1%	0.4761
categorized	CATwebProfundidadPag, CATcantOcurrencias, CATcatPalPagina, CATlongPalabra, CATcantVocalesFuerte, CATcantVocalesDebile, CATlongOracion, CATcantNumeros, CATcantSignosEspeciales	9	63.82%	0.3043
nominal	tema, tipoPag, palAntTipo, paisRadicacion, terminacion, siguePuntuacion , clasePag, empiezaMayuscula, resaltada, esTitulo, fraseEspecial	12	65.63%	0.34
independent	palAntTipo, tipoPag, siguePuntuacion resaltada, CATlongPalabra, CATcantVocalesFuerte CATcantVocalesDebile, CATlongOracion	8	63.51%	0.3

As can be seen from the results in Table 6, there is a low correctly-classified rate and kappa values.

4) Instance windowing

Three windows of instances were selected. The windows were of different size and composition as described below:

a)sample 1: 47829 instances. The word-class distribution is: 6689 nouns, 2762 verbs, 11027 other class, 36 unknown class. Main characteristics of the sample: words were extracted from pages mainly with the same subtopic within the set theme. Besides, each page were longer than in the other two samples.

b)sample 2: 20515 instances. The word-class distribution is: 6392 nouns, 3050 verbs, 11054 other class, 19 unknown class. Main characteristics of the sample: pages were related to many different subtopics and typically very short in the average.

c)sample 3: 20524 instances. The word-class distribution is: 6535 nouns, 2954 verbs, 11014 other class, 21 unknown class. Main characteristics of the sample: pages were related to different subtopics but with intermediate size in the average.

The model training was performed with each sample, taking 12 data fields (4 of them categorical). Results are shown in Table 7.

Table 7. Results with Different samples

set	correctly classified	Kappa statistic
1	71.1%	0.4740
2	76.8%	0.6052
3	76.9%	0.5986

As can be seen from the table, there is a significant variation of classification power with the dataset. Those results are due the characteristic of each one. As a consequence of these characteristics, the noun rate is highest in the second sample, making the classification correctness higher than sample 1 and lower than sample 3. Kappa statistics decreases for sample 3, which has a fewer number of nouns than sample 2, even considering that sample 3 performs a bit better classification rate due to the shorter pages.

3.2. Classification Using descriptors and Stemming

The classifier behavior was studied considering stem. Sample 2 was extended with the corresponding radixes using stemming algorithm. Records with same stem were counted and those whose stem frequency is lower than 10 were eliminated from the set. The resulting set has 2316 instances.

Classification model was constructed with distinct attribute considerations: several simple global descriptors, stem and three simple descriptors, stem and six simple descriptors, stem as unique descriptor. Table 8 shows the results obtained: correctly classified rate improves with stemming combined with descriptors. Kappa value denotes that it is a better model also (κ increases up to 0.887). It can be seen that the field stem is not as good for classification by kind of word (tipoPal) as descriptors do.

Table 8. Stem with/without Descriptors

case	correctly classified	Kappa statistic
no stem	64.4%	0.4000
stem and 3 fields	90.7%	0.8500
stem and 6 fields	93.0%	0.8870
stem alone	02.5%	0.0168

3.3. Word Classification with Stemming and best Morpho-Syntactical Descriptors

The 12 best descriptors (4 of them were categorical) are selected and combined with syntactical-radixes. Such descriptors describe the topic of the document, kind of word, kind of html page, kind of previous word, word suffix, word length, number of vowels, etc.
Results with and without stemming are shown in Table 9. Here the confidence level has improved very much when considering stem.

Table 9 Descriptors with/without Stem

	with stem	without stem
correctly classified	94%	64%
Kappa statistic	0.90	0.40

4. CONCLUSIONS AND FUTURE WORK

From the previous sections some interesting conclusions can be extracted:
-Training set must have more than 20514 to get better results.
-Categorization procedure takes influence on the classifier confidence, improving it when the number of categories increases.
-The best subset of data fields have many interdependencies.
-The html-page length influences the performance. Better results are obtained with lengthy pages.
-Stem has not enough classification power by itself.
-Descriptors have not enough classification power by itself.
-A combination of stemming with detected better descriptors makes it possible to perform word classifications with good confidence levels.

Some interesting future works are:
-Repeat this analysis considering as kind of previous words: “none”, “article”, “preposition”, “pronoun”, and “other”.
-Analyze categorical field dependencies to reduce the number of variables with a kind of formula.
-Study the variations due to other field categorization criteria.
-Evaluate alternate algorithms.
-Compare results against other sources as books, magazines, etc.

REFERENCES

[1]. Alani H. et al. (2003) "Automatic Extraction of Knowledge from Web Documents", In Proc. of 2nd International Semantic Web Conference - Workshop on Human Language Technology for the Semantic Web and Web Services, Sanibel Island.

[2]. Aldezábal I. (1996) "Del analizador morfológico al etiquetador: unidades léxicas complejas y desambiguación". Procesamiento del lenguaje natural. N. 19, pp. 90-100. España.

[3]. Díaz Villa A.M. (2005) "Tipología de errores gramaticales para un corrector automático" Magazine Procesamiento del Lenguaje Natural, vol 35.

[4]. Fernández Lanza S. (2003) "Una contribución al procesamiento automático de sinonimia utilizando Prolog" PhD dissertation, Santiago de Compostela, España.

[5]. Figuerola C. G. (2000) "Categorización automática de documentos en español: algunos resultados experimentales". ReLIS, Jornadas de Bibliotecas Digitales. http://imhotep.unizar.es/jbidi/jbidi2000/14_2000.pdf

[6]. Genthial D. (1990) "Contribution of a Category Hierarchy ti the Robustness of Syntacic Parsing", 13th CoLing, vol. 2, pp. 139-144. Helsinki, Finland.

[7]. Gulla A. A. (1996) "A Sign Expansion Approach to Dynamic, Multi-purpose Lexicons", International Conference on Computational Linguistics. Proceedings of the 16th Conference on Computational Linguistics. Vol. 1. pp. 478 – 483. Copenhagen. Denmark.

[8]. Levinson S. (2006) "Statistical Modeling and Classification", AT&T Bell Laboratories, Murray Hill, New Jersey, USA. Also available at http://cslu.cse.ogi.edu/HLTSurvey/ch11node4.html.

[9]. Manning C., Schütze H. (1999) "Foundations of Statistical Natural Language Processing", Cambridge, Mass. MIT Press. ISBN 0262133601

[10]. Mateo P.L., González J.C., Villena J., Martínez J.L. (2003) "Un sistema para resumen automático de textos en castellano" DAEDALUS S.A., Madrid, España.

[11]. Mitchell T. (1997) Machine Learning, New York: WCB/Mc Graw Hill, pp. 51-80.

[12]. Nießen S., Ney H. (2000) "Improving SMT quality with morpho-syntactic analysis", in Proc. of the 18th conference on Computational linguistics – Vol. 2, pp. 1081 – 1085, Saarbrücken, Germany.

[13]. Oliveira O.N., Nunes M.G. V., Oliveira M.C. F. (1998) "Por qué no podemos hablar con una computadora?" Magazine of Sociedad Mexicana de Física., México, v. 12, pp. 1 - 9.

[14]. Platzer C., Dustdar S. (2005) "A Vector Space Search Engine for Web Services", in Proc. of the Third European Conference on Web Services (ECOWS' 05), Vaxjo, Sweden.

[15]. Porter, M. F. (1980) "An Algorithm for suffix Stripping", Program, vol. 14 (3), pp. 130-137.

[16]. Seretan V., Nerima L., Vehrli E. (2004) "Using the Web as a Corpus for the Syntactic-Based Collocation Identification", in Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, pages 953-960, Sydney, Australia.

[17]. Trabalka M., Bieliková M. (2000) "Using XML and Regular Expressions in the Syntactic Analysis of Inflectional Language", In Proc. of Symposium on Advances in Databases and Information Systems (ADBIS-DASFAA'2000), Praha. pp. 185-194

[18]. Witten I. H., Frank E. (2005) "DataMining – Practical Machin Learning Tools and Techniques", 2nd ed., San Francisco: Morgan Kaufmann Publishers.