NATO Science for Peace and Security Series
D: Information and Communication Security - Vol. 19

# Mining Massive Data Sets for Security

## Advances in Data Mining, Search, Social Networks and Text Mining, and their Applications to Security

Edited by
Françoise Fogelman-Soulié
Domenico Perrotta
Jakub Piskorski
Ralf Steinberger

IOS
Press

This publication is supported by: The NATO Science for Peace and Security Programme

ISBN 978-1-58603-898-4

9 781586 038984

www.iospress.nl

ISBN 978-1-58603-898-4
ISSN 1874-6268

**NATO Science for Peace and Security Series**

This Series presents the results of scientific meetings supported under the NATO Programme: Science for Peace and Security (SPS).

The NATO SPS Programme supports meetings in the following Key Priority areas: (1) Defence Against Terrorism; (2) Countering other Threats to Security and (3) NATO, Partner and Mediterranean Dialogue Country Priorities. The types of meeting supported are generally "Advanced Study Institutes" and "Advanced Research Workshops". The NATO SPS Series collects together the results of these meetings. The meetings are co-organized by scientists from NATO countries and scientists from NATO's "Partner" or "Mediterranean Dialogue" countries. The observations and recommendations made at the meetings, as well as the contents of the volumes in the Series, reflect those of participants and contributors only; they should not necessarily be regarded as reflecting NATO views or policy.

**Advanced Study Institutes** (ASI) are high-level tutorial courses to convey the latest developments in a subject to an advanced-level audience.

**Advanced Research Workshops** (ARW) are expert meetings where an intense but informal exchange of views at the frontiers of a subject aims at identifying directions for future action.

Following a transformation of the programme in 2006 the Series has been re-named and re-organised. Recent volumes on topics not related to security, which result from meetings supported under the programme earlier, may be found in the NATO Science Series.

The Series is published by IOS Press, Amsterdam, and Springer Science and Business Media, Dordrecht, in conjunction with the NATO Public Diplomacy Division.

**Sub-Series**

| | | |
|---|---|---|
| A. | Chemistry and Biology | Springer Science and Business Media |
| B. | Physics and Biophysics | Springer Science and Business Media |
| C. | Environmental Security | Springer Science and Business Media |
| D. | Information and Communication Security | IOS Press |
| E. | Human and Societal Dynamics | IOS Press |

http://www.nato.int/science
http://www.springer.com
http://www.iospress.nl

Sub-Series D: Information and Communication Security – Vol. 19 ISSN 1874-6268

# Mining Massive Data Sets for Security

## Advances in Data Mining, Search, Social Networks and Text Mining, and their Applications to Security


Edited by

Françoise Fogelman-Soulié

*KXEN*

Domenico Perrotta

*European Commission – Joint Research Centre*

Jakub Piskorski

*European Commission – Joint Research Centre*

and

Ralf Steinberger

*European Commission – Joint Research Centre*



IOS Press

Amsterdam • Berlin • Oxford • Tokyo • Washington, DC

Published in cooperation with NATO Public Diplomacy Division

Proceedings of the NATO Advanced Study Institute on Mining Massive Data Sets for Security,
Gazzada (Varese), Italy
10–21 September 2007



ISBN 978-1-58603-898-4
Library of Congress Control Number: 2008935667

*Publisher*
IOS Press
Nieuwe Hemweg 6B
1013 BG Amsterdam
Netherlands
fax: +31 20 687 0019
e-mail: order@iospress.nl

*Distributor in the UK and Ireland*
Gazelle Books Services Ltd.
White Cross Mills
Hightown
Lancaster LA1 4XS
United Kingdom
fax: +44 1524 63232
e-mail: sales@gazellebooks.co.uk

*Distributor in the USA and Canada*
IOS Press, Inc.
4502 Rachael Manor Drive
Fairfax, VA 22032
USA
fax: +1 703 323 3668
e-mail: iosbooks@iospress.com



PRINTED IN THE NETHERLANDS

# Contents

**Social Networks**



**Text Mining**



**Security Applications**

*Mining Massive Data Sets for Security*
*F. Fogelman-Soulié et al. (Eds.)*
*IOS Press, 2008*

*doi:10.3233/978-1-58603-898-4-245*



# Using linguistic information as features for text categorization

Arturo MONTEJO-RÁEZ [1], Luis Alfonso UREÑA-LÓPEZ,
Miguel Ángel GARCÍA-CUMBRERAS and José Manuel PEREA-ORTEGA
*University of Jaén, Spain*

**Abstract.** We report on some experiments using linguistic information as additional features as part of document representation. The use of linguistic features on several information retrieval and text mining tasks is a hot topic, due to the polarity of conclusions encountered by several researchers. In this work, extracted information of every word like the *Part Of Speech*, *stem* and *morphological root* have been combined in different ways for experimenting on a possible improvement in the classification performance and on several algorithms. Our results show that certain gain can be obtained when these varied features are combined in a certain manner, and that these results are independent from the set of classification algorithms applied or the evaluation paradigm chosen, providing certain consistency to our conclusions in text categorization on the Reuters-21578 collection.

**Keywords.** Automatic text categorization, linguistic features, document representation

## Introduction

We report on some experiments using linguistic information as additional features in a classical Vector Space Model [1]. Extracted information of every word like the *Part Of Speech* and *stem*, *morphological root* have been combined in different ways for experimenting on a possible improvement in the classification performance and on several algorithms, like SVM[2], BBR[3] and PLAUM.

The inclusion of certain linguistic features as additional data within the document model is being a subject of debate due to the variety of conclusions reached. This work exposes the behavior of a text categorization system when some of these features are integrated. Our results raise several open issues that should be further studied in order to get more consistent conclusions on the subject. Linguistic features may be useful or not depending on the task, the language domain, or the size of the collection. Nevertheless, we focus here on a very specific aspect: the way we combine features is also crucial for testing its effectiveness.

Automatic Text Classification (TC), or Automatic Text Categorization as it is also known, tries to relate documents to predefined set of classes. Extensive research has been carried out on this subject [4] and a wide range of techniques are applicable to solve this task: feature extraction [5], feature weighting, dimensionality reduction [6], machine

[1] Corresponding Author: Universidad de Jaén, Jaén 23071, Spain; E-mail: amontejo@ujaen.es.

learning algorithms and more. Besides, the classification task can be either binary (one out of two possible classes to select), multi-class (one out of a set of possible classes) or multi-label (a set of classes from a larger set of potential candidates). In most cases, the latter two can be reduced to binary decisions [7], as the algorithm used does in our experiments [8]. This is the reason why machine learning algorithms have been playing a central role in TC.

In order to do machine learning when dealing with documents, a proper representation of the document has to be built. So far, the most common strategy is to follow the *bag of words* approach, where words from the document are extracted, transformed in some way and then weighted according to their frequency of use within the document. In this manner, documents are represented as vectors, where each dimension corresponds to the weight of a given term (i.e. a lemmatized word or a multi-word mainly) in the document.

Due to the large amount of terms within any vocabulary, reduction strategies must be applied in order to reduce the dimensionality of these document vectors. For dimension reduction there are also several solutions, which we can broadly classify into two main approaches: feature selection and feature transformation. The former relies upon mechanisms that discard non relevant features in some way [5], [6], [9], while the second one is related to methods using representation in reduced dimension feature spaces, such as term clustering approaches [10] or Latent Semantic Indexing [11].

This work focuses on the early phase of document representation, deciding which information from the document is extracted as features. In a step forward to the bag of words, we study how some of the output data that we can obtain from Natural Language Processing (NLP) methods can enrich document representation by evaluating a text categorization problem as a proof of concept.

## 1. Considering linguistic features

In Natural Language Processing, the document is a source of valuable information related to the different levels of analysis that can be performed on a given text. Nowadays, several linguistic tools are available for analyzing our documents content and extracting lexical and syntactic information, along with emerging and more abstract information at semantic level. Some of the information that can be considered as available from text by applying NLP could be the morphological root of word (e.g. *construct* as replace for *constructed*; more examples in table1), a multi-word term (e.g. noun phrases like *tropical plant*), the resolution of anaphora (e.g. *Sara was playing cards with John and she asked him to leave* could be replaced by *Sara was playing cards with John and Sara asked John to leave*), part-of-speech (POS) analysis (e.g. *I (Pronoun) told (Verb) you (Pronoun)*), semantic roles, dependency trees as result of shallow parsing, and named entities (e.g. *United Nations* as a unique term).

Our hypothesis is that adding data from a higher level of abstraction will enrich our feature space with additional information whenever this data is related in some way. We believe this is due to the fact that information derived from base data by more abstracted reasoning incorporates new information, as that reasoning is performed on heuristics and knowledge beyond the scope of the problem domain (i.e. the explicit content of the document). That is, the knowledge behind NLP tools is aggregated to new features and should, therefore, be exploited by the system.

| original word | morphological root | stem |
|---|---|---|
| communications | communication | commun |
| decided | decide | decid |
| becoming | become | becom |
| bought | buy | bought |

**Table 1.** Examples of obtained stems and morphological roots

Now the question is: *how to incorporate this abstract information, to the Salton's Vector Space Model in a blind way?* We can find previous research on applying NLP to text categorization successfully in the work by Sable, McKeown and Church [12], but their method is based on a careful consideration and combination of linguistic features. Our concern is on adding some linguistic features as additional information into a traditional bag-of-words representation with no further processing. Of course, every possible combination of linguistic features is not considered here. Our goal is rather to prove that some of them could lead to certain enhanced versions of document representation. This assertion argues against some previous related work, like the one by Moschitti and Basili [13], but is consistent with the conclusions given by Bigert and Knutsson [14] and Pouliquen et al [15]. In this last work, the authors explored the possible benefits of incorporating stop-word removal, multi-word detection and lemmatisation, concluding that these were very limited in the case of multi-word treatment and lemmatization, but a remarkable one when eliminating stop-words.

Moschitti and Basili's research [13] incorporates POS tags, noun senses and complex nouns (multi-words) as features for text categorization. These enriched document representations have been generated and tested on Reuters-21578[2], Ohsumed[3] and 20-NewsGroups[4] benchmark collections. They found worthless improvements. We think that some possible combinations were missing, while in our research such combinations are studied.

## 2. Experiments

In this section, the algorithm applied for multi-label classification is introduced along with the description of the data preparation phase and the results obtained in the designed experiments.

### 2.1. Multi-label classifier system

In the *Adaptive Selection of Base Classifiers* (ASBC) approach [16] we basically train a system using the battery strategy (many classifiers working together independently), but *(a)*, we allow tuning the binary classifier for a given class by a balance factor, and *(b)* we provide the possibility of choosing the best of a given set of binary classifiers. To this end, the algorithm introduces a hyper-parameter $\alpha$ parameter resulting in the algorithm given in figure 1. This value is a threshold for the minimum performance allowed to a binary classifier during the validation phase in the learning process, although the class

[2] http://www.daviddlewis.com/resources/testcollections/reuters21578/

[3] http://trec.nist.gov/data/filtering/

[4] http://people.csail.mit.edu/jrennie/20Newsgroups/

still enters into the evaluation computation. If the performance of a certain classifier (e.g. F1 measure, described in next section) is below the value $\alpha$, meaning that the classifier performs badly, we discard the classifier and the class completely. By doing this, we may decrease the recall slightly (since less classes get trained and assigned), but we potentially may decrease computational cost, and increase precision. The effect is similar to that of the *SCutFBR* [17]. We never attempt to return a positive answer for rare classes. In [16], it is shown how this filtering saves us considering many classes without important loss in performance.

---

Input:
  a set of training documents $D_t$
  a set of validation documents $D_v$
  a threshold $\alpha$ on the evaluation measure
  a set of possible label (classes) $L$,
  a set of candidate binary classifiers $C$
Output :
  a set $C' = \{c_1, ..., c_k, ..., c_{|L|}\}$ of trained binary classifiers
Pseudo code:
  $C' \leftarrow \emptyset$
  for-each $l_i$ in $L$ do
    $T \leftarrow \emptyset$
    for-each $c_j$ in $C$ do
      *train-classifier*$(c_j, l_i, D_t)$
      $T \leftarrow T \cup \{c_j\}$
    end-for-each
    $c_{best} \leftarrow$ *best-classifier*$(T, D_v)$
    if *evaluate-classifier*$(c_{best}) > \alpha$
      $C' \leftarrow C' \cup \{c_{best}\}$
    end-if
  end-for-each

---

**Figure 1.** Adaptive Selection of Base Classifiers algorithm

The binary base classifiers selected within our experimental framework have been: Support Vector Machines (SVM) [2] under its implementation in the SVM-Light package[5], Logistic Bayesian Regression [3] using the BBR software[6] and the Perceptron Learning Algorithm with Uneven Margins [18] implemented natively in the TECAT package (which itself implements the whole ASBC multi-label strategy)[7]. All base classifiers have been configured with default values.

[5] Available at `http://svmlight.joachims.org/`
[6] Available at `http://www.stat.rutgers.edu/ madigan/BBR/`
[7] Available at `http://sinai.ujaen.es/wiki/index.php/TeCat`

**Table 2.** Contingency Table for $i$ Category

| | YES is correct | NO is correct |
|---|---|---|
| YES is assigned | $a_i$ | $b_i$ |
| NO is assigned | $c_i$ | $d_i$ |

### 2.2. Evaluation Measures

The effectiveness of a classifier can be evaluated with several known measures [22]. The classical "Precision" and "Recall" for Information Retrieval are adapted to the case of Automatic Text Categorization. From categorizing test documents using a trained system, a contingency table is completed (Table 2), and then the precision and recall are calculated following equations 1 and 2.

$$P_i = \frac{a_i}{a_i + b_i} \tag{1}$$

$$R_i = \frac{a_i}{a_i + c_i} \tag{2}$$

On the other hand, the precision and recall can be combined using the $F_1$ measure:

$$F_1(R, P) = \frac{2PR}{P + R} \tag{3}$$

In order to measure the average performance of a system, three measures can be used: micro-averaged precision $P_\mu$, macro-averaged precision in a document basis $P_{macro-d}$ and macro-averaged precision in a category basis $P_{macro-c}$.

$$P_\mu = \frac{\sum_{i=1}^{K} a_i}{\sum_{i=1}^{K} (a_i + c_i)} \tag{4}$$

$$P_{macro} = \frac{\sum_{i=1}^{K} P_i}{K} \tag{5}$$

where $K$ is the number of categories or the number of documents depending on the basis used.

Recall and F1 measures are computed in a similar way. In our experiments we have used these measures in order to prove the effectiveness of the studied system.

### 2.3. Data preparation

The data used was the "ModApte" split of the Reuters-21578[8] collection, a dataset well known to the research community devoted to text categorization problems [19]. This collection contains 9,603 documents in the training set, while the test set is composed

[8]Prepared by David D. Lewis. The collection is freely available from the web page http://www.research.att.com/~lewis/reuters21578.html

of 3,299 documents. Each document is assigned to an average of slightly more than 2 classes. Documents contain little more than one hundred words per document.

In order to verify the contribution of the new features, we have combined them to be included into the vector space model by preprocessing the mentioned collection through some of the analysis tools available in the GATE architecture[9] [20]. Thus, we have generated enriched collections in the following ways:

1. `word` (w): a corpus with just plain text without any additional parsing has been used as base case
2. `stem` (s): each word has been transformed by applying the classical Porter's Stemmer algorithm [21]
3. `root` (r): instead of words, we consider their lexical roots
4. `stem+POS` (s+p): stems are, in this corpus, attached to their identified Part-Of-Speech, thus, each feature is a pair `stem-POS` (represented in our naming convention by a "+" sign)
5. `word+POS` (w+p): every word is attached to the associated POS tag
6. `root+POS` (r+p): every lexical root is attached to the associated POS tag
7. `word-root-stem-pos` (w-r-s-p): finally, a corpus every all previous features are in the document as independent features

### 2.4. Results

When evaluating text categorization, micro-averaged measures have been traditionally chosen as indicators of system quality. In multi-label text categorization we could also consider the possibility of using two additional indicators: macro-averaged measures by document and macro-averaged measures by class. These two are totally different and depending on how we want to apply our system, this choice may be crucial to really understand the performance of a proposed solution. In this way, macro-averaged precision by document, for instance, will tell us about how precise the labels are that we assign to every single document. On the other hand, macro-averaged precision by class will tell us how precise we are in assigning classes to documents in general. Certain differences arise since most of the classes are normally seldom assigned to most of the documents (there are many rare classes in real classification systems). Therefore, macro-averaging by document is an interesting indicator when the system is intended for individual document labeling. Of course, the counterpoint here is that if we are good with most frequent classes, then macro-averaged measurements by document will report good results, hiding bad behavior on rare classes, even when rare classes may be of higher relevance, since they are better discriminators when labels are used for practical matters. In our study, these three evaluation paradigms have been included.

In tables 3, 4 and 5, F1, precision and recall measurements on all the experiments run are shown. The best results obtained according to the algorithm used have been highlighted in cursive. The results in bold represent the feature combination that reported best performance on each algorithm and each of the three evaluation paradigms considered.

We can draw some conclusions from these evaluation measurements. The main one, that the winning feature combination turned out to be *w-r-s-p*. The use of the morphological root performs better than using stemming in general, although without noticeable

[9] Available at `http://gate.ac.uk`

| F1 | *w* | *r* | *r+p* | *s* | *s+p* | *w+p* | *w-r-s-p* |
|---|---|---|---|---|---|---|---|
| **SVM avg** | 0.8211 | 0.8302 | 0.8224 | 0.8283 | 0.8234 | 0.8233 | **0.8358** |
| **SVM dAVG** | 0.8040 | 0.8212 | 0.8065 | 0.8194 | 0.8060 | 0.8086 | **0.8268** |
| **SVM cAVG** | 0.4345 | 0.4984 | 0.4673 | 0.4979 | 0.4979 | 0.4637 | **0.5208** |
| **BBR avg** | 0.8323 | 0.8367 | 0.8358 | 0.8305 | 0.8345 | 0.8323 | **0.8384** |
| **BBR dAVG** | *0.8323* | 0.8367 | 0.8358 | *0.8305* | 0.8345 | 0.8323 | **0.8384** |
| **BBR cAVG** | 0.4972 | 0.5696 | 0.5201 | 0.5648 | 0.5134 | 0.5046 | **0.5759** |
| **PLAUM avg** | *0.8337* | *0.8392* | *0.8388* | *0.8323* | *0.8384* | *0.8392* | ***0.8412*** |
| **PLAUM dAVG** | 0.8238 | *0.8375* | *0.8362* | 0.8253 | *0.8376* | *0.8376* | ***0.8392*** |
| **PLAUM cAVG** | *0.5323* | *0.6015* | *0.5531* | *0.5842* | *0.5528* | *0.5460* | ***0.6126*** |

**Table 3.** Combined F1 measurements on different algorithms and feature sets

| Precision | *w* | *r* | *r+p* | *s* | *s+p* | *w+p* | *w-r-s-p* |
|---|---|---|---|---|---|---|---|
| **SVM avg** | *0.9277* | *0.9150* | *0.9226* | *0.9147* | *0.9269* | *0.9263* | *0.9212* |
| **SVM dAVG** | 0.8195 | 0.8364 | 0.8220 | 0.8354 | 0.8219 | 0.8253 | **0.8420** |
| **SVM cAVG** | 0.6933 | 0.7302 | 0.6997 | 0.7302 | 0.7176 | 0.7034 | **0.7614** |
| **BBR avg** | **0.9204** | 0.8956 | 0.9068 | 0.8873 | 0.9065 | 0.9107 | 0.9022 |
| **BBR dAVG** | 0.8348 | **0.8450** | 0.8421 | *0.8393* | 0.8400 | 0.8380 | 0.8441 |
| **BBR cAVG** | 0.7583 | 0.7948 | 0.7594 | 0.8005 | 0.7585 | 0.7420 | ***0.8170*** |
| **PLAUM avg** | **0.9142** | 0.8935 | 0.8992 | 0.9014 | 0.8959 | 0.9016 | 0.8937 |
| **PLAUM dAVG** | *0.8368* | *0.8469* | *0.8472* | 0.8366 | *0.8474* | *0.8477* | *0.8476* |
| **PLAUM cAVG** | 0.7532 | *0.7997* | *0.7804* | *0.8008* | *0.7718* | *0.7679* | **0.8139** |

**Table 4.** Combined precision measurements on different algorithms and feature sets

| Recall | *w* | *r* | *r+p* | *s* | *s+p* | *w+p* | *w-r-s-p* |
|---|---|---|---|---|---|---|---|
| **SVM avg** | 0.7364 | 0.7598 | 0.7418 | 0.7569 | 0.7407 | 0.7410 | **0.7650** |
| **SVM dAVG** | 0.8033 | 0.8223 | 0.8064 | 0.8199 | 0.8050 | 0.8075 | **0.8277** |
| **SVM cAVG** | 0.3448 | 0.4113 | 0.3777 | 0.4097 | 0.3783 | 0.3707 | **0.4280** |
| **BBR avg** | 0.7596 | **0.7851** | 0.7752 | *0.7806* | 0.7730 | 0.7663 | 0.7830 |
| **BBR dAVG** | 0.8228 | **0.8444** | 0.8362 | *0.8396* | 0.8337 | 0.8302 | 0.8413 |
| **BBR cAVG** | 0.3996 | 0.4800 | 0.4301 | 0.4766 | 0.4234 | 0.4122 | **0.4848** |
| **PLAUM avg** | *0.7663* | *0.7911* | *0.7860* | 0.7730 | *0.7878* | *0.7849* | ***0.7946*** |
| **PLAUM dAVG** | *0.8279* | *0.8458* | *0.8440* | 0.8307 | *0.8466* | *0.8447* | ***0.8477*** |
| **PLAUM cAVG** | *0.4412* | *0.5220* | *0.4691* | *0.4999* | *0.4676* | *0.4598* | ***0.5359*** |

**Table 5.** Combined recall measurements on different algorithms and feature sets

performance differences. This can explain why people still apply stemming algorithms, which are easier to implement. Categorization results do not seem to improve when using stems and roots as replacement for words without morphological normalization, although they are useful to reduce the feature space. On the other side, when combined, categorization performance improves. This makes us think that there exist synergistic dependencies among them.

In order to validate these observations, statistical significance has been computed by applying a two-tailored Wilcoxon test on the obtained results. This test is the non-parametric equivalent of the paired samples *t-test*. This implies the assumption that both distributions are symmetrical, in which case the mean and medians are identical. Thus, the null hypothesis (usually represented by $H_0$) considers that for the two distributions the median difference is zero.

Distributions have been generated for each feature combination and for each evaluation measure. Thus, at each evaluation measure we have 60 values (3 algorithms multiplied by 30, the measurements obtained for the 30 most frequent categories). In tables 6, 7, 8 we have the p-values obtained using the two-tailored signed rank test (Wilcoxon test) comparing each possible pair of feature combinations. Values related to statistically significant differences are shown in bold (i.e. those p-values below 0.05).

| *Precision* | w | r | s | w+p | r+p | s+p | w-r-s-p |
|---|---|---|---|---|---|---|---|
| w | 0.50000000 | 0.99975792 | 0.99999722 | 0.99795972 | 0.99996494 | 0.99871684 | 0.99879193 |
| r | **0.00024208** | 0.50000000 | 0.99120325 | **0.00191301** | 0.21569861 | 0.08772633 | 0.28198043 |
| s | **0.00000278** | **0.00879675** | 0.50000000 | **0.00025781** | **0.02299721** | **0.01308712** | **0.01383293** |
| w+p | **0.00204028** | 0.99808699 | 0.99974219 | 0.50000000 | 0.94710365 | 0.78149820 | 0.97230034 |
| r+p | **0.00003506** | 0.78430139 | 0.97700279 | 0.05289635 | 0.50000000 | **0.01874444** | 0.58880332 |
| s+p | **0.00128316** | 0.91227367 | 0.98691288 | 0.21850180 | 0.98125556 | 0.50000000 | 0.83800353 |
| w-r-s-p | **0.00120807** | 0.71801957 | 0.98616707 | **0.02769966** | 0.41119668 | 0.16199647 | 0.50000000 |

**Table 6.** Two-tailored Wilcoxon test over Precision

| *Recall* | w | r | s | w+p | r+p | s+p | w-r-s-p |
|---|---|---|---|---|---|---|---|
| w | 0.50000000 | **0.00000004** | **0.00000041** | **0.03389618** | **0.00003132** | **0.00013093** | **0.0000000001** |
| r | 0.99999996 | 0.50000000 | 0.69496983 | 0.99999625 | 0.99945972 | 0.99992046 | 0.21925151 |
| s | 0.99999959 | 0.30503017 | 0.50000000 | 0.99998501 | 0.99785479 | 0.99985558 | 0.07531379 |
| w+p | 0.96610382 | **0.00000375** | **0.00001499** | 0.50000000 | **0.00019374** | **0.00856058** | **0.00000009** |
| r+p | 0.99996868 | **0.00054028** | **0.00214521** | 0.99980626 | 0.50000000 | 0.59073375 | **0.00002613** |
| s+p | 0.99986907 | **0.00007954** | **0.00014442** | 0.99143942 | 0.40926625 | 0.50000000 | **0.00000146** |
| w-r-s-p | 1.00000000 | 0.78074849 | 0.92468621 | 0.99999991 | 0.99997387 | 0.99999854 | 0.50000000 |

**Table 7.** Two-tailored Wilcoxon test over Recall

| *F1* | w | r | s | w+p | r+p | s+p | w-r-s-p |
|---|---|---|---|---|---|---|---|
| w | 0.50000000 | **0.00005713** | **0.00071825** | 0.30361848 | **0.00924403** | **0.00698185** | **0.00000046** |
| r | 0.99994287 | 0.50000000 | 0.95565518 | 0.99969308 | 0.99808699 | 0.99932684 | 0.16276175 |
| s | 0.99928175 | **0.04434482** | 0.50000000 | 0.99728397 | 0.98571432 | 0.99709336 | **0.01274590** |
| w+p | 0.69638152 | **0.00030692** | **0.00271603** | 0.50000000 | **0.00760854** | **0.01334894** | **0.00000191** |
| r+p | 0.99075597 | **0.00191301** | **0.01428568** | 0.99239146 | 0.50000000 | 0.29375643 | **0.00016408** |
| s+p | 0.99301815 | **0.00067316** | **0.00290664** | 0.98665106 | 0.70624357 | 0.50000000 | **0.00002037** |
| w-r-s-p | 0.99999954 | 0.83723825 | 0.98725410 | 0.99999809 | 0.99983592 | 0.99997963 | 0.50000000 |

**Table 8.** Two-tailored Wilcoxon test over F1

Regarding precision, the use of the original text without processing is the best option. But in terms of recall and F1, root and stem features may be preferred. Although root and *w-r-s-p* combination show similar results, from the p-value of the second one over the first one, we can observe that *w-r-s-p* is close to overperform root with statisticall significance.

## 3. Conclusions and future work

Our results show that certain linguistic features improve the categorizer's performance, at least on Reuters-21578. A text classification system shows many degrees of freedom (different tuning parameters), and small variations can produce big deviations, but from the results above, it is clear that for any of the algorithms selected and on any of the evaluation paradigms, the feature combination *word-root-stem-pos* produces better results, but with small improvements compared to the other feature combinations, like morphological root, according to the F1 measure.

Though the gain in precision and recall is not impressive, we believe that further research has to be carried out in this direction, and we plan to study different integration strategies, also considering additional features like *named entities*, term lists and additional combinations of all these features in the aim of finding more synergy. Also, the impact of such information may be higher for full texts than short fragments of Reuters-21578 texts. Collections like the HEP [23] or the JRC-Acquis [24] corpora will be used to analyze this possibility.

At this final point, we would like to underline relevant issues regarding the usage of linguistic features that should also be studied. Some languages (Slavonic languages and Finno-Ugric) are more highly inflected, i.e. there are more variations for the same lemma than, for example, in English. Another important issue is the trade-off between possible errors in the generation of these features by the linguistic tools used and the benefit that their inclusion can produce on the final document representation. Word sense disambiguation may introduce more noise into our data. Also, the stemming algorithm, may perform badly in texts of specialized domains and may harm the final categorization results. Finally, the size of the collection, the length of the document and other characteristics of the data can determine whether the inclusion of certain features is useful or not. Therefore, many questions remain open and the research community still has work to do on this topic.

## Acknowledgements

This work has been partially financed by the TIMOM project (TIN2006-15265-C06-03) granted by the Spanish Ministry of Science and Technology and the RFC/PP2006/Id_514 granted by the University of Jaén.

## References

[1] Gerard Salton, A. Wong, and C. S. Yang. A Vector Space Model for Automatic Indexing. Technical Report TR74-218, Cornell University, Computer Science Department, July 1974.

[2] Thorsten Joachims. Text categorization with support vector machines: learning with many relevant features. In Claire Nédellec and Céline Rouveirol, editors, *Proceedings of ECML-98, 10th European Conference on Machine Learning*, number 1398, pages 137–142, Chemnitz, DE, 1998. Springer Verlag, Heidelberg, DE.

[3] D. Madigan, A. Genkin, D. D. Lewis, and D. Fradkin. Bayesian multinomial logistic regression for author identification. In *25th International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering*, volume 803 of *AIP Conference Proceedings*. American Institute of Physics, August 2005.

[4] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM Comput. Surv.*, 34(1):1–47, 2002.

[5] D. D. Lewis. Feature Selection and Feature Extraction for Text Categorization. In *Proceedings of Speech and Natural Language Workshop*, pages 212–217, San Mateo, California, 1992. Morgan Kaufmann.

[6] Daphne Koller and Mehran Sahami. Toward optimal feature selection. In *International Conference on Machine Learning*, pages 284–292, 1996.

[7] Erin L. Allwein, Robert E. Schapire, and Yoram Singer. Reducing multiclass to binary: A unifying approach for margin classifiers. In *Proc. 17th International Conf. on Machine Learning*, pages 9–16. Morgan Kaufmann, San Francisco, CA, 2000.

[8] A. Montejo-Ráez and L.A. Ureña López. Binary classifiers versus adaboost for labeling of digital documents. *Sociedad Española para el Procesamiento del Lenguaje Natural*, (37):319–326, 2006.

[9] Yiming Yang and Jan O. Pedersen. A comparative study on feature selection in text categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning*, pages 412–420. Morgan Kaufmann Publishers Inc., 1997.

[10] L. Douglas Baker and Andrew Kachites McCallum. Distributional clustering of words for text classification. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 96–103. ACM Press, 1998.

[11] Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by Latent Semantic Analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

[12] C. Sable, K. McKeown, and K. Church. Nlp found helpful (at least for one text categorization task). In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Philadelphia, PA., 2002.

[13] Alessandro Moschitti and Roberto Basili. Complex linguistic features for text classification: A comprehensive study. In *ECIR*, pages 181–196, 2004.

[14] Johnny Bigert and Ola Knutsson. Robust error detection: A hybrid approach combining unsupervised error detection and linguistic knowledge. In *Proceedings of Romand 2002 (Robust Methods in Analysis of Natural language Data)*, Frascati, Italy, July 2002.

[15] Bruno Pouliquen, Ralf Steinberger, and Camelia Ignat. Automatic Annotation of Multilingual Text Collections with a Conceptual Thesaurus. In Amalia Todirascu, editor, *Proceedings of the workshop 'Ontologies and Information Extraction' at the EuroLan Summer School 'The Semantic Web and Language Technology'(EUROLAN'2003)*, page 8 pages, Bucharest (Romania), 2003.

[16] A. Montejo-Ráez, R. Steinberger, and L. A. Ureña López. Adaptive selection of base classifiers in one-against-all learning for large multi-labeled collections. In Vicedo J. L. et al., editor, *Advances in Natural Language Processing: 4th International Conference, EsTAL 2004*, number 3230 in Lectures notes in artifial intelligence, pages 1–12. Springer, 2004.

[17] Yiming Yang. A study on thresholding strategies for text categorization. In W. Bruce Croft, David J. Harper, Donald H. Kraft, and Justin Zobel, editors, *Proceedings of SIGIR-01, 24th ACM International Conference on Research and Development in Information Retrieval*, pages 137–145, New Orleans, US, 2001. ACM Press, New York, US. Describes RCut, Scut, etc.

[18] Y. Li, H. Zaragoza, R. Herbrich, J. Shawe-Taylor, and J. Kandola. The perceptron algorithm with uneven margins. In *Proceedings of the International Conference of Machine Learning (ICML'2002)*, 2002.

[19] F. Debole and F. Sebastiani. An analysis of the relative hardness of reuters-21578 subsets. 2004.

[20] H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan, and Y. Wilks. Experience of using GATE for NLP R&D. In *Proceedings of the Workshop on Using Toolsets and Architectures To Build NLP Systems at COLING-2000*, Luxembourg, 2000. `http://gate.ac.uk/`.

[21] M. F. Porter. *An algorithm for suffix stripping*, pages 313–316. Morgan Kaufmann Publishers Inc., 1997.

[22] D. D. Lewis. Evaluating Text Categorization. In *Proceedings of Speech and Natural Language Workshop*, pages 312–318. Morgan Kaufmann, 1991.

[23] A. Montejo-Ráez. *Automatic Text Categorization of Documents in the High Energy Physics Domain*. PhD thesis, University of Granada, March 2006.

[24] Ralf Steinberger, Bruno Pouliquen, Anna Widiger, Camelia Ignat, Tomaž Erjavec, Dan Tufiş, and Dániel Varga. The JRC-Acquis: a multilingual aligned parallel corpus with 20+ languages. *The 5th International Conference on Language Resources and Evaluation (LREC'2006)*, pages 2142–2147, May 2006.