Empirical investigation of fast text classification over linguistic features

Roberto Basili and **Alessandro Moschitti** and **Maria Teresa Pazienza** [1]

Abstract. Recently, an original extension of the well-known Rocchio model (i.e. the Generalized Rocchio Formula (GRC)) as a feature weighting method for text classification has been presented. The assessment of such a model requires a statistically motivated parameter estimation method and wider empirical evidence. In this paper, three different corpora have been adopted in two languages. Results suggest that GRC, integrating linguistic information, is a viable more efficient alternative to state-of-art TC systems.

1 Introduction

Methods for integrating linguistic content within information retrieval activities are receiving a growing attention [10]. Work in text retrieval through the Internet suggests that embedding linguistic information at a suitable level within traditional quantitative approaches is useful to bring the experimental stage to operational results. This is a representational problem. In this paper traditional methods for statistical text categorization augmented by a systematic use of linguistic information are analysed. The purpose is to determine linguistic information useful to improve text classification (TC) accuracy. This deserves of extensive evaluation over heterogeneous resources and data sets. In TC a systematic experimental framework is possible: tasks and performance factors, sensitive to the available linguistic information, can be assessed and measured over well-assessed benchmarking data sets.

In [2] an original extension of the well-known Rocchio model (i.e. the Generalized Rocchio Formula (GRC)) as a feature weighting method for TC has been proposed. Parameterized weighting was observed as a suitable technique to improve the impact of more informative features on the TC accuracy. The benefits are mainly due to the tuning the Rocchio formula parameters. Sensitivity of the formula to different values of the parameters has been also discussed in [5], where warnings on the estimation methodology are also raised. The learning and classification efficiency of Generalized Rocchio text classifiers (GRC) make them appealing for operational scenarios, e.g. Web document classification. The size of data sets prevents the adoption in these cases other effective machine learning approaches, e.g. Support Vector Machine [7] and k-Nearest Neighbor [13]. The lower TC accuracy is the major drawback of Rocchio classifiers. Nevertheless the GRC has been shown to significantly improve the basic performances of simpler Rocchio models on benchmarking data, ([2]). Now, it is crucial to design and test general and effective parameter estimation methods for GRC. In this perspective some issues need explanation:

- A systematic analysis of general properties of the training model. The Reuters benchmark used in [2] (with a fixed split between training and test data) may represent a bias[2] for the results. A systematic testing is needed to show that the parameter estimation does not depend on the documents chosen for training and testing.
- Assessment of performance figures. The widely used Breakeven Point depends on acceptance thresholds that are adjusted until precision is equal to recall. It thus depends on testing data and may not reflect the real system performances. f-measure is a more accurate performance index.
- Larger evidence over different corpora. Corpus that relates to different domains, (e.g. more close to real scenario), may have different behaviors. Note that in [2] only the Reuters benchmark has been used which has a more academic nature than realistic data.
- Multilingual evidence. Different languages may result in different TC behavior. As English is successfully approached via stemming (as a way to capture word information) this is not always true for morphologically richer languages (e.g. Italian).

The aims of this paper are thus:

- First, define a statistical parameter estimation technique for the Generalized Rocchio classifier (GRC) and test it over different corpora. More general results have been derived over three different domains: News from Reuters collection, medical documents from Ohsumed corpus and news in Italian from ANSA, the main Italian news agency. A rigorous text sampling for held-out performance evaluation has been applied to derive reliable results.
- Second, measure the role of available linguistic information. In particular, syntactic characterization (via POS tagging) and multiwords expression (i.e. terms derived from corpora) have been adopted as features on the available corpora. Measures of their impact on classification performances have been obtained.

In Section 2, the basic problem of TC and the linguistic framework used for feature extraction are described. The selection model based on the generalized Rocchio formula with its weighting capabilities is presented in Section 3 where the parameter estimation procedure is also defined. In Section 4 experiments are reported aiming to show the effectiveness of the proposed estimation technique as well as to quantify the contribution of linguistic information.

2 Text Classification and feature extraction

The classification problem is the derivation of a decision function that maps documents into one or more target classes, $C =$

[1] University of Rome Tor Vergata Department of Computer Science Systems and Production 00133 Roma (Italy) {basili, moschitti, pazienza}@info.uniroma2.it

[2] The splitting between test and training related to a version of Reuters corpus is a specific partition. It is thus possible it does not represent the full properties of the corpus

$\{C_1, ..., C_n\}$, representing topics (e.g. "*Politics, Entertainment*"). An extensive collection of texts already classified, often called *training set*, induces the classification function.

Profile-based (or linear) classifiers are characterized by a function based on a similarity measure between the representation of incoming documents d and each class C_i. Both representations are vectors and similarity is traditionally estimated as the cosine angle between the two vectors. The description $\vec{C}_i$ of each target class (C_i) is usually called *profile*, that is the vector summarizing the content of all the training documents, i.e. those pre-categorized under C_i. The vector components are called *features* and refer to independent dimensions in the space in which similarity is estimated. The i-th components of a vector representing a given document d is a numerical weight associated to the i-th feature w of the dictionary that occurs in d. Similarly, profiles are derived from the grouping of positive instances d in class C_i, i.e. $d \in C_i$.

Traditional techniques (e.g. [12]) make use of single words w as basic features. In the next section the kind of linguistic information used to define class and document vectors as well as the processes used to extract them are described.

2.1 Linguistic features in text categorization

Linguistic content in TC can be emphasized by determining $features$ able to express complex textual evidences for the classification function. Basic language processing capabilities traditionally allow to extend the knowledge about words occurring in documents, like for example their canonical forms (i.e. the morphological derivation from a lemma) and their syntactic roles (i.e. part-of-speech (POS) in the input context). Previous works on text classification [9, 11] suggest that availability of significant complex sequence of terms increases the indexing performances. Recognition of Proper Nouns and extraction of terminological expressions from texts are Natural Language Processing (NLP) techniques able to discover and match in documents linguistically motivated sequences of words. The aim, here, is to verify if such high-level descriptors are in fact better indexes for TC.

2.1.1 The extraction of linguistic features

The extraction of complex syntactic structures is allowed by a robust $Parser$, i.e. a linguistic processor that takes a normalized version of documents and produces a set of grammatical and semantic information for each text ([3]). The information produced by the parser includes:

- Lemmas or multiwords expressions. Simple words (e.g. $bank$ and $match$) or functional expressions (e.g. *in order to* and *as well as*) are detected.
- Proper Nouns (PNs). In line with systems for Information Extraction, Named-Entities are recognized by extensive catalogs as well as by the application of NE grammars. A typed set of proper nouns is derived from each news and processed independently from the other lemmas.
- Syntactic Categories of lemmas. Units of text (i.e. simple or complex terms) are tagged by a single Part-of-Speech (POS), (e.g. `N` for nouns, `V` for verbs). Document descriptions include lemmas with their own POS, so that verbal and nominal occurrences are independent (e.g. $rate$/`V` $\neq$ $rate$/`N`).

Besides the above information, features include also *terminological expressions* that are automatically derived from the source corpora (i.e. training documents). They are complex nominal expressions that will be defined in the next Section.

2.1.2 Corpus-driven terminology extraction

The derivation of terminological noun phrases is supported by an inductive method for (off-line) terminology extraction (TE), early introduced in [1]. It is based on an integration of symbolic and statistical modeling. First, relevant atomic terms ht (i.e. singleton words) are identified by traditional techniques, e.g. the idf score early suggested in [12]. Linguistically principled grammars[3] are then applied to identify linguistic structures (headed by ht). They are admissible candidates for terminological expressions. Finally, extracted candidates are validated and selected by the use of statistical filters. Statistical properties imposed on the occurrences of multiword sequences aim to restrict the semantic relations expressed by terms. Only the early recognized expressions that are validated statistically are retained as legal terminological entries of the underlying domain. Several methods of corpus-driven TE have been proposed. We adopted for our tests the methods detailed in [1].

In terminology terms are surface canonical forms of structured expressions referring to entities with complex properties in a domain. They are nouns or noun phrases generally denoting specific concepts in a given corpus, i.e. in a given domain.

Usually term candidates are couples $(x, \vec{y})$, where $\vec{y}$ represents the sequence of (left and/or right) modifiers, e.g. *(disk, (-1,hard))*, *(system, ((-2,cable),(-1,television))* for *hard disk* and *cable television system*, respectively. Among a number of statistical filters *mutual information* (MI), $I(x, y) = \log_2 \frac{P(x,y)}{P(x)P(y)}$, has been often used to capture linguistic associations (e.g. [4]).

In TE, MI can be reliably computed over two words. n-ary relations (e.g. *federal securities laws*, *Federal Home Loan Bank board* require the estimation of joint probabilities of n words. The need of a huge amount of data for a reliable outcome for n-ary relations usually impacts over data sparseness problems. Thus an approach based on binary MI to collections of words has been used:

$$I(x, \vec{y}) = \log_2 \frac{P(x,\vec{y})}{P(x)P(\vec{y})}$$

where the conceptual link is considered between word x and the vector $\vec{y} = (y_1, y_2, ..., y_n)$. Thresholding over MI $I(x, \vec{y})$, as detailed in [1], provides a straightforward and effective decision criteria.

According to the above method specific terminological datasets, $Term_i$ are derived from training texts available for the class C_i. During parsing, items in $\cup_i Term_i$ will thus matched and included in the document features. Notice how the terminological database ($TDB = \cup_i Term_i$) is derived automatically for each collection so that the TC experiments described in the Section 4, will make use of three different TDBs.

3 Extending the Rocchio's formula for optimal feature selection and weighting

Feature selection relates to the application of statistical methods (information gain, χ^2, mutual information ...), for pruning non relevant features. Major drawbacks are that features irrelevant for a class may be removed even if they are important for another one [7]. The problem here is to give the right weight to a given feature in different classes for determining its impact. A feature selection strategy based on a machine learning algorithm seems a promising methodology.

[3] The parser supports the phases (e.g. tokenization, Part-of-Speech tagging and lemmatization) for the grammatical recognition of term structures.

The Rocchio's formula has such a property. It has been traditionally used in Profile-Based Text Classification and is defined as follows. Given:

- the set of training documents R_i classified under the topics C_i (positive examples),
- the set $\bar{R}_i$ of the documents not belonging to C_i (negative examples) and
- ω_f^h, the weights[4] of feature f in document h,

the weight Ω_f^i of a given feature f in the profile of the class $C_i =< \Omega_f^1, \Omega_f^2, ... >$ is:

$$\Omega_f^i = \max\left\{0, \frac{\beta}{|R_i|}\sum_{h\in R_i}\omega_f^h - \frac{\gamma}{|\bar{R}_i|}\sum_{h\in \bar{R}_i}\omega_f^h\right\} \quad (1)$$

In Eq. 1 the parameters β and γ control the relative impact of positive and negative examples and determine the weight of f in the i-th profile[5].

As noticed in [2], the relevance of a feature deeply depends on the corpus characteristic and, in particular, on the differences among the training material for the different classes, e.g. size, the structure of topics or the style of documents. They sensibly change according to text collections and classes. Moreover, it provides a rather smooth feature selection. Features are used only when they influence the similarity estimation for all and only the classes for which they are selective. The γ and β setting allows to drastically limit noise without direct feature elimination. For this Eq. 1 provides scores, Ω_f^i, that have been directly used as weights in the associated feature space ([2]). Each category has, in this way, its own set of relevant and irrelevant features. The optimal values of these two parameters are also estimated independently for each class i. This results in a vector of (γ_i,β_i) couples each one optimizing the performance of the classifier over the i-th class. For each class, one parameter (β_i=1) is fixed and γ_i can be tuned. The weighting, ranking and selection scheme used is thus the following:

$$\Omega_f^i = \max\left\{0, \frac{1}{|R_i|}\sum_{h\in R_i}\omega_f^h - \frac{\gamma_i}{|\bar{R}_i|}\sum_{h\in \bar{R}_i}\omega_f^h\right\} \quad (2)$$

From now on we will refer to this model as the GRC classifier. Equation 2 has been applied given the parameters γ_i that for each class C_i lead to the maximum Breakeven point on C_i (see [13] for more details on BEP).

3.1 Estimating parameters in a generalized Rocchio model

The idea of parameter adjustment in the Rocchio formula is not completely new. In [5] it has been pointed out that these parameters greatly depend on the training corpus and different settings of their values produce a significant variation in performances. However their estimation was not clarified. The major problem was that the simple parameter estimation procedure that provides the lowest $training$ set error produced a small improvement in the error rate over the reference test-set. The reason relies in the corpus adopted (i.e. Reuters 21478). It has a fixed splitting between training and test sets with quite different distributions of categories. This prevents the possibility of correctly estimating effective γ values from training data. The erroneous conclusion that the parameters are a property of the document set used for their estimation was derived. As it will be shown in Section 4 this is not true.

When a random splitting between learning LS and test TS data is allowed, the following parameter estimation for Eq. 2 can be carried out according to a held-out estimation procedure:

1. First, a subset of LS, called estimation set ES, is defined.
2. The set $LS - ES$ is then used for profile building
3. Estimation of the γ_i parameters is finally carried out over ES.

Performance of the resulting model can be thus measured over the TS documents. Notice that this procedure can be applied iteratively if steps 2-3 are carried out according to different, randomly generated splits ES_k and $LS-ES_k$. Several vectors $\vec{\gamma}_i$ are thus derived at steps k, denoted by $\vec{\gamma}_i^{(k)}$. A final $\vec{\Gamma}_i$ can be thus obtained via a point wise estimator Θ applied to distribution of $\vec{\gamma}_i^{(k)}$, i.e.

$$\vec{\Gamma}_i = \Theta(\vec{\gamma}_i^{(1)}, ..., \vec{\gamma}_i^{(K)}) \quad (3)$$

Performance of the model parameterized by Γ_i can be then measured over the TS documents.

The above procedure is easily applicable whenever the number of documents in the training set LS is large enough for ES (or ES_k) to be representative of all the classes. If the number of training documents available in ES for a class C_i is too low, the parameter estimation procedure that optimizes BEP is not stable, possibly producing biased results.

4 Performance Evaluation

Experiments have two aims: to asses the general performances of GRC (Eq. 2) driven by the estimation procedure of Section 3.1 and to evaluate the contribution of the linguistic features presented in Section 2.1. Performance indexes are derived via a cross validation technique applied as follows:

- Generate $n = 30$ random splits of the corpus: about 70% for training (LS) and 30% for testing (TS).
- For each split σ
  - Learn the classifier on LS^σ. In case of GRC apply the estimation procedure of Section 3.1. This leads to sampling the sets $ES^\sigma{}_k$ for the estimation of γ_i.
  - Evaluating performance indexes on TS_σ
  - For the BEP index evaluation an algorithm to find the value of recall equal to precision is applied.
  - In case of f-measure computation thresholds are estimated over the $ES^\sigma{}_k$, as it is done for parameters γ_i.
- The final indexes are the mean BEPs and f-measures values, derived from each split, according to:

$$\bar{BEP} = \frac{\sum_\sigma BEP(TS_\sigma)}{n} \qquad \bar{f_1} = \frac{\sum_\sigma f_1(TS_\sigma)}{n}. \quad (4)$$

4.1 The experimental set-up

With the aim to obtain more general results three different collections have been considered. The first reference collection is the Reuters corpus, version 3, prepared by Apté [13][6]. The collection includes

[4] Several methods are used to assign weights of a feature, as widely discussed in [12].

[5] In [6], Eq. (1) has been used with values β = 16 and γ = 4 for the categorization of low quality images.

[6] A formatted version of this collection was prepared by Y. Yang and colleagues, and is currently available at Carnegie Mellon University's web site through http://moscow.mt.cs.cmu.edu:8081/reuters 21450/apte.

11,098 documents for 93 classes, with a fixed splitting between test TS and learning data LS (3,309 vs. 7,789). In this work TS and LS have been merged so that random splits are derived. The Reuters collection, used in many experiments (e.g. [13, 7]) supports comparative analysis.

The second collection (Ohsumed, at `ftp://medir.ohsu.edu/pub/ohsumed`), compiled by William Hersh, includes 50,216 medical abstracts. The first 20,000 have been used in all our experiments. The target classes are the 23 *MeSH diseases* categories.

The third corpus includes about 15,000 news in Italian from the ANSA agency. It refers to 8 target categories (with a size of approximately 2,000 documents). ANSA categories relate to typical newspaper contents (e.g. Politics, Sport, Economy)[7]. It is worth to note that this last collection is closer to operational scenarios. It suffers from a human error in corpus preparation: some documents are not correctly assigned to the categories and other ones are repeated more then once.

Performance scores are always expressed by means of *Breakeven point* ($B\bar{E}P$) and *f-measure* with equal importance assigned to recall and precision ($\bar{f}_1$). The global performance of a systems is given by the *microaveraging* that refers to all categories contained in the target corpus. The *Token* feature set includes unstemmed words that do not appear in the $SMART$ stop list. The linguistic feature sets have been built including POS-tagged lemmas and terminological expressions. These last are derived from available training material independently for each class. For example, in the TDB of the class acq (i.e. $Mergers\ and\ Acquisition$) of the Reuters corpus, among the 9,650 different features about 1,688 are made of terminological expressions or proper nouns (17%). The weight ω_f^h of a feature f in a document h is the usual product between the logarithm of the frequency of f in h and the associated inverse document frequency.

4.2 Cross evaluation of Generalized Rocchio Classifier

In the first set of experiments the traditional Rocchio classifier has been evaluated to determine the base-line performances. Two different values for its parameterization have been selected from literature [7, 6, 5]. The sets of features used in these experiments are all tokens (i.e. no feature selection has been applied), which are about 39,000 for Reuters, 42,000 for Ohsumed and 55,000 for ANSA. The novelty of these sets is the inclusion of numbers and the words composed of special characters. We prefer to use all these features because they result in higher performances (one or two percent point wrt the stems). In the second set of experiments the estimation procedures of parameters and thresholds for GRC have been applied. 20 random samples for estimating thresholds and γ_i have been extracted from the current training set. Each sample contains about 30-40% of training documents.

The performances are reported in tables 1, 2 and 3 respectively for Reuters, Ohsumed and ANSA. The column labeled with *Rocchio* refers to the original classifier parameterized with $\gamma = 0.4$ and $\gamma = 1$. The column GRC expresses two performance indexes (f-measure and BEP) for the generalized Rocchio Classifier. The difference between f_1 and BEP measures (on average) the complexity of estimating the thresholds. The results, for each corpora, assess the benefit of the generalized Rocchio formula used together with the parameter estimation procedure. An improvement of about 3-4% over the simple Rocchio is observed wrt the global Microaverage performances (evaluated by using all categories of the target corpus). In the tables we can also observe the relevant improvement for some categories[8]. Moreover, in order to better locate the GRC performances wrt other literature works, we tested the Support Vector Machine (SVM) classifier [7] over our feature sets. The last column in Tables 1 and 2 reports the results for the linear version of SVM. We observe that GRC outcome is very close to SVM which is considered the current state-of-art.

Table 1. Performance comparisons of Generalized Rocchio classifier on Reuters corpus

	Rocchio (BEP)		GRC		SVM
Category	$\gamma = 0.4$	$\gamma = 1$	BEP	f_1	f_1
earn	95.20	95.23	95.17	95.39	98.80
acq	80.91	84.38	86.35	86.12	96.97
money-fx	73.34	75.37	77.80	77.81	87.28
grain	74.71	78.02	88.74	88.34	91.36
crude	83.44	83.23	83.33	83.37	87.16
trade	73.38	74.80	79.39	78.97	79.13
interest	65.30	68.63	74.60	74.39	82.19
ship	78.21	80.53	82.87	83.17	88.27
wheat	73.15	75.57	89.07	87.91	83.90
corn	64.82	66.60	88.01	87.54	83.57
Microav.(93 cat.)	80.07	81.50	84.90	84.42	88.30

Table 2. Performance Comparisons of Generalized Rocchio classifier on Ohsumed corpus

	Rocchio (BEP)		GRC		SVM
Category	$\gamma = 0.4$	$\gamma = 1$	BEP	f_1	f_1
Pathology	37.57	47.06	48.78	50.58	48.55
Cardiovascular	71.71	75.92	77.61	77.82	80.79
Immunologic	60.38	63.10	73.57	73.92	72.89
Neoplasms	71.34	76.85	79.48	79.71	80.16
Digestive Systems	59.24	70.23	71.50	71.49	71.10
Microav.(23 cat.)	54.36	61.79	66.06	65.81	68.37

It is worth noting that our results about SVM on Reuters are higher than those found in literature [7]. This is not surprising given the higher dimension of the feature space used here. In [7] only 10,000 stems have been used for experiments and *numbers* were likely removed. This class of tokens has a relevant role in Reuters corpus as also explained in [8]. The results in [7] for Reuters corpus are about 2 percent points lower than those reported in Table 1 (for both Rocchio and SVM). Thus SVM and GRC have higher performances by using all possible tokens.

Table 3. Performance comparisons of Generalized Rocchio classifier on ANSA corpus

	Rocchio (BEP)		GRC	
Category	$\gamma = 0.4$	$\gamma = 1$	BEP	f_1
News	50.35	61.06	69.80	68.99
Economics	53.22	61.33	75.95	76.03
Foreign Economics	67.01	67.08	65.09	61.72
Foreign Politics	61.00	67.23	75.80	75.59
Economic Politics	72.54	80.52	78.66	68.95
Politics	60.19	67.49	60.07	59.58
Entertainment	75.91	78.14	77.64	77.63
Sport	67.80	78.98	80.00	80.14
Microaverage	61.76	69.23	72.36	71.00

[7] This corpus is used within the HLT European project NAMIC that includes ANSA

[8] Only some categories are reported in tables. We have shown those classes useful for comparisons.

4.3 Evaluation of linguistic contribution

In these experiments the feature sets which include the POS of a word and complex terminological expressions (terms) have been used for training GRC. The global performances in Tables 4, 5 and 6 show small improvements wrt the bag of words. However, if we look at the individual category performance, we observe several classes that take significant advantages from linguistic material. In Reuters, most of the categories (shown in Table[9]) reach higher performances.

In Ohsumed corpora only some categories increase the performances on linguistic data. The ANSA corpora seems more sensible than the others as some (overspecialized) categories, like spe (entertainment) or pec (economical politic), show higher accuracy.

Table 4. Linguistic contribution to the Generalized Rocchio classifier performances on Reuters corpus

	Tokens	Terms	Terms+POS
Category	f_1	f_1	f_1
earn	95.39	95.40	95.25
acq	86.12	87.83	87.46
money-fx	77.81	79.03	79.04
grain	88.34	87.90	87.89
crude	83.37	83.54	83.47
trade	78.97	79.72	79.59
interest	74.39	75.93	76.05
ship	83.17	83.30	83.42
wheat	87.91	87.37	86.76
corn	87.54	87.87	87.32
Microav.(93 cat.)	84.42	84.97	84.82

The good results on some categories are due to the use of NLP methods that allows to include as features *n-gram* not bound to a specific n. Terminological expressions may span over more than 2 or 3 constituents: complex proper nouns like *Federal Home Loan Bank* are usually captured. More interestingly, chains of noun phrases modifying other nouns or even proper nouns, as in *federal securities laws*, *temporary restraining order*, *Federal Home Loan Bank board* are recognized and included in the feature set. However their inclusion in category profiles have to be carried out carefully. As a side effect, they change the weights of other features (e.g. the simple tokens). This is the major reason for performance decreases. The use of separate features space could be a solution for making operative the use of linguistic information in TC.

Table 5. Linguistic contribution to the Generalized Rocchio classifier performances on Ohsumed corpus

	Tokens		Terms	
Category	f_1	BEP	f_1	BEP
Pathology	48.78	50.58	49.36	51.13
Cardiovascular	77.61	77.82	77.48	77.74
Immunologic	73.57	73.92	73.51	74.03
Neoplasms	79.48	79.71	79.38	79.77
Dig. Systems	71.50	71.49	71.28	71.46
Hemic. & Lymph.	65.80	65.75	65.93	65.85
Neonatal	50.05	49.98	52.83	52.71
Skin	60.38	60.59	60.53	60.80
Nutr. & Metab.	60.08	60.20	60.66	60.75
Endocrine	44.80	48.76	43.96	48.87
Env. Disorder	64.54	64.58	64.92	64.98
Animal	34.35	38.02	37.39	39.45
Microav.(23 cat.)	65.81	66.06	65.90	66.32

5 Conclusion

In this paper, a robust model for fast text categorization and its training procedure have been described and tested. The proposed estimation procedure of the GRC parameters defines a systematic feature selection and weighting technique. The experimentation of GRC on very different corpora (Reuters, Ohsumed and ANSA) have conclusively shown that it is robust and effective with respect to noise and ambiguity in the data. Performances are close to systems with a significantly higher learning complexity (e.g. SVM). GRC can be thus considered as a better alternative for operational scenarios like Web classification and searching.

Table 6. Linguistic contribution to the Generalized Rocchio classifier performances on ANSA corpus

	Tokens	Terms	Terms+POS
Category	f_1	f_1	f_1
News	68.99	68.58	69.30
Economics	76.03	75.21	75.39
Foreign Economics	61.72	61.12	62.37
Foreign Politics	75.59	75.32	76.36
Economic Politics	68.95	75.78	76.89
Politics	59.58	62.48	63.43
Entertainment	77.63	76.48	76.27
Sport	80.14	79.63	79.67
Microaverage	71.00	71.80	72.37

Natural language techniques have been adopted to improve TC accuracy. In particular complex nominal structure have been used in place of the usual n-grams. Increases of performances are not stably observed although it does not prevent the adoption of NLP capabilities for efficient TC. The higher performances shown on some categories suggest that linguistic structures capture relevant although not critical information. Further ways for exploiting them in TC is part of our future research.

[9] Note that they are the 10 top sizes of the Reuters Corpora

REFERENCES

[1] R. Basili, G. De Rossi, and M.T. Pazienza, 'Inducing terminology for lexical acquisition', in *Preoceeding of EMNLP 97 Conference, Providence, USA*, (1997).

[2] R. Basili, A. Moschitti, and M.T. Pazienza, 'NLP-driven IR: Evaluating performances over text classification task', in *Proceedings of IJCAI 2001 Conference, Seattle, USA*, (2001).

[3] Roberto Basili, Maria Teresa Pazienza, and Fabio Massimo Zanzotto, 'Efficient parsing for information extraction', in *Proc. of the ECAI98*, Brighton, UK, (1998).

[4] K. W. Church and P. Hanks, 'Word association norms, mutual information, and lexicography', *Computational Linguistics*, **16**(1), (1990).

[5] William W. Cohen and Yoram Singer, 'Context-sensitive learning methods for text categorization', *ACM Transactions on Information Systems*, **17**(2), 141–173, (1999).

[6] David J. Ittner, David D. Lewis, and David D. Ahn, 'Text categorization of low quality images', in *Proceedings of SDAIR-95*, pp. 301–315, Las Vegas, US, (1995).

[7] Thorsten Joachims, 'Text categorization with support vector machines: Learning with many relevant features.', in *In Proceedings of ECML-98*, pp. 137–142, (1998).

[8] Sofus A. Macskassy, Haym Hirsh, Arunava Banerjee, and Aynur A. Dayanik, 'Using text classifiers for numerical classification', in *Proceeding of IJCAI-01*, ed., Bernhard Nebel, Seattle, US, (2001).

[9] Dunja Mladenić and Marko Grobelnik, 'Word sequences as features in text-learning', in *Proceedings of ERK-98, the Seventh Electrotechnical and Computer Science Conference*, pp. 145–148, Ljubljana, SL, (1998).

[10] Dan I. Moldovan and Rada Mihalcea, 'Using wordnet and lexical operators to improve internet searches', *IEEE Internet Computing*, **January-February**, (2000).

[11] Bhavani Raskutti, Herman Ferrá, and Adam Kowalczyk, 'Second order features for maximising text classification performance', in *Proceedings of ECML-01*, (2001).

[12] G: Salton and C. Buckley, 'Term-weighting approaches in automatic text retrieval.', *Information Processing and Management*, **24(5)**, 513–523, (1988).

[13] Y. Yang, 'An evaluation of statistical approaches to text categorization', *Information Retrieval Journal*, (1999).