An automatic SVM-based strategy for Digital Protocol [1]

Rosita Guido [1], Michelangelo Misuraca [2], Francesca Vocaturo [2]

[1] D.E.I.S. - Univ. della Calabria - 87036 Arcavacata di Rende (CS) - Italy

[2] D.E.S. – Univ. della Calabria - 87036 Arcavacata di Rende (CS) - Italy

Abstract

The main goal of Digital Protocol in Public Administrations is to make the document flows more functional and efficient, and to improve the transparency of the administrative actions. Commercial solutions for automatic document routing frequently do not respond to the standards of official protocols defined by the law. On the other hand many Digital Protocol platforms implemented at the present require both a wide human control and a specific expertise. In this paper an automatic strategy based on Support Vector Machine is proposed aiming at analysing and classifying documents with respect to their textual content.

Keywords: digital protocol, factorial analysis, support vector machine

1. Introduction

Information overload is one of the critical states in businesses and institutions. Many human resources have been usually spent for sorting out by hand the most useful documents, but this is often a difficult and time-consuming activity. The computer management of document flows is an essential tool within the more general process of designing and developing automated information systems, and it plays a strategic role to support a policy of services improving and cost containment.

Since 2004 in Italy the digitalisation of Official Protocols in the different Public Administrations is mandatory. The main aspiration of the different laws has been to make the document flows more functional and efficient. Furthermore the possibility of accessing to the different procedure steps by the interested subjects has pointed out, aiming at improving the transparency of the administrative action.

Several commercial solutions have been developed for the automatic routing of documents, but they frequently do not respond to the standards of official protocols defined by law. Moreover, many platforms use at most semiautomatic procedures so that human resources are still needed, particularly in the training step. Manual category label assignment is a tedious and slow task and automatic approaches are more and more demanded.

Text Categorization (TC) is a main topic in Text Mining (TM). We are interested in labelling the documents in a collection, from a knowledge discovery standpoint, having the aim of satisfying a specific informative need. In this sense the term "categorization" is used to mean

[1] This work has been financially supported by the research grant "*Analisi dei Dati Testuali e Classificazione Automatica di Documenti nella Pubblica Amministrazione*" (Fondo ex60% 2009, Università della Calabria).

the automatic matching between each document and one or more categories. This task can be seen, in a more general TM process, as a first step for selecting the most useful information with respect to an analysed phenomenon. In a second step it is possible to use this information for grouping the documents via an automatic classification procedure.

In this work a dataset of documents, manually selected from a bigger collection, is pre-treated and analysed with the aim of automatically classifying incoming new documents on the basis of their textual content. An interesting way for dealing with this task is the use of a supervised classification technique. Supervised approaches attempt to discover and represent the relations between a set of input attributes and a defined target through a learning step, where a training set of data is analysed aiming at finding a certain classification. Once a classifier is defined it is tested on another set of data and validated in terms of accuracy. In the following a real case study is introduced and the use of a particular class of classifiers based on Support Vector Machine (SVM) is evaluated.

2. Digitalising an Official Protocol

The problem of classifying the large amount of documents that is everyday produced or received by a Public Administration has been dealt with, during the years, more from a juridical than a technical point of view [2]. With the digital revolution of the last 20 years also in an official framework has become possible to acquire and preserve each type of document in electronic format, even if the document management still requires a human intervention.

Nowadays, with the rapid development of information technologies and web-based services, the private and public organizations have to handle large amounts of information expressed as classical qualitative and quantitative data as well as textual data. In this frame the task of digitalising different documents has to take into account not only the filing but also the whole knowledge management process.

In this paper the attention is focused on a peculiar step of such process, concerning automatic classification (without any, or at least a weak, human control) of new incoming documents on the basis of the textual information.

2.1. Problem definition

The starting point is the Digital Protocol used by the Department of Linguistics of the University of Calabria (Italy). The average flow of incoming documents is 300 pages per month. At present, a coding system of 101 labels grouped in 7 macro-categories (namely, *titolario*) is used, and the codes are manually imputed via a software platform originally projected and developed for the specific requirements of the Department. The document routing is automatic, so that the system directly sends the registered documents to the interested subjects.

The language used in the documents is the typical bureaucratic language used by Public Administrations. Generally the terms are not semantically ambiguous, so that the main source of ambiguity comes from the different lexical role of the homographs in the text. The latter aspect can be handled by recurring to a lexicalization of repeated segments or by using a *part of speech* tagging. Conversely, because of the closeness among some categories in terms of characterizing words (*e.g.*, the categories *Fatture* and *Acquisti* can share keywords like *prezzo*, *quantità* and *iva*), it can be not trivial to univocally define a specific vocabulary for each category.

[2] DPR 445/2000; DPCM 31/10/2000; AIPA memorandum n. 28/2001; AIPA resolution n. 42/2001.

A collection of 647 documents received by the Department between March and June 2008 has been considered. Each document has been manually labelled with the Department coding system by a trained registrar employer. Because of the original paper format and the available PDF format of the documents, the collection has been parsed via an OCR software.

In this stage of the research, it has been necessary to eliminate any document written in a language different from Italian (a very frequent case in this kind of Department) and the documents with personal identifiable information. The resulting *corpus* contains 430 documents labelled with 8 categories (Tab. 1).

Category	*N. of documents*
Acquisti	35
Direttore	256
Fatture	103
Fin. e contabilità	6
Iva	4
Resp. progetto	9
Mandati	12
Personale	5

Table 1: Categorization of the 430 documents

A first pre-processing step has been necessary for avoiding the scanning noises and errors and obtaining more significant data. The normalized *corpus* has a vocabulary of 19.685 distinct terms, with a *type*/*token* ratio of 14%.

A preliminary lexical analysis of the documents has been performed, having an explorative standpoint, for defining the specific vocabulary of each category. In this way it is possible to characterize the different documents classes with respect to their textual content. An example for the documents labelled as *Direttore* is shown in Tab. 2, where "internal %" and "global %" represent the number of occurrences of each keyword with respect to the number of occurrences of the keywords used in the given category and in the whole *corpus*, respectively.

Keyword	*Internal %*	*Global %*	*Test-value*	*Probability*
progetto	1.10	0.79	0.907	0.182
bando	0.88	0.63	0.655	0.256
attività	0.88	0.63	0.655	0.256
presentazione	0.88	0.63	0.655	0.256
prestazione	0.66	0.47	0.357	0.361
comunicazione	0.66	0.47	0.357	0.361
…	…	…	…	…

Table 2: Specific vocabulary of the category Direttore

At the end of the pre-process procedures a lexical table *documents* × *words* has been obtained, by considering a set of 2032 keywords with a number of occurrences greater than 4.

2.2. Explorative Analysis

From a statistical point of view it is possible to analyse a lexical table by considering two different perspectives: investigating the similarities among documents in terms of shared vocabulary to highlight the presence of groups, and/or investigating the language used in the different documents to discover topics or concepts.

Several factorial methods have been developed in different scientific domains, *e.g.* Latent Semantic Indexing (LSI) and Correspondence Analysis (CA). However, it is remarkable that all these techniques share common algebraic properties (Balbi and Misuraca, 2006). Factorial reduction methods are traditionally used to synthesize a dataset and visualize it in a bi-dimensional subspace. In a statistical frame such methods aim at decomposing the lexical table and showing latent association relationships among documents (or words) in the space spanned by words (or documents).

In this work a Lexical Correspondence Analysis has been performed on the 430 documents in order to represent the relations between the different categories. A technique based on a weighted Euclidean metric such as CA seems to be more suitable because of the different length of documents and the presence of rare but meaningful keywords.

In Fig. 1 the association structure of the analysed documents in the keywords space is reported. The different colours (in gray scale) show the different 8 categories.

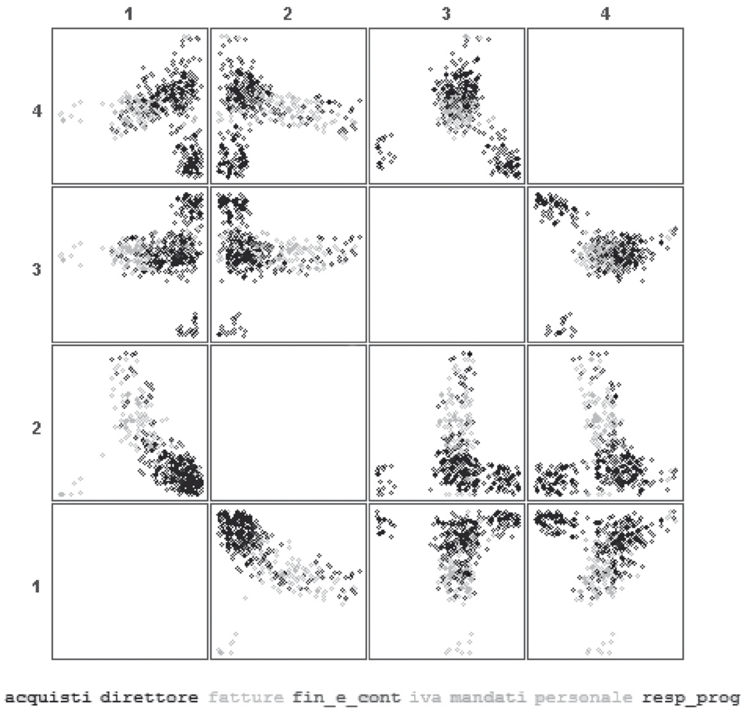

Figure 1: Graphical representation of the first 4 factorial maps

Even if the amount of total inertia explained by each factorial map is not really considerable, because of the well known problems of factorial methods on very sparse matrices, it is possible to note a certain dependence structure pointed out by the documents with respect to the shared vocabulary.

3. Theoretical Framework

Text Categorization (TC) can be defined as the task of assigning a Boolean value to each pair (d_i,c_j) of D×C, where D is a set of documents and C is a set of pre-defined categories. A value 1 is assigned if document d_i is classified under a category c_j, while a value 0 is assigned if d_i is not classified under c_j. The aim is therefore to approximate an unknown target function

φ^*: $D \times C \rightarrow \{0,1\}$ that describes how documents should be classified with a function φ: $D \times C \rightarrow \{0,1\}$, called *classifier*, so that it coincides "as much as possible" with φ^*.

Generally a problem of classification can be approached by considering the use of unsupervised or supervised techniques. The main difference is that unsupervised approaches require very few subjective choices into the classification processes, and a natural grouping of documents is obtained on the basis of a pre-defined criterion. This feature is really suitable when no previous additional information or expert knowledge is available. Supervised approaches are instead based on a learning process where a labelled set of data is used for training classifiers, then a testing dataset is usually considered for validating the results.

When a large amount of new documents have to be classified the analysis has to be performed again because the relations between incoming documents and the clusters are unknown. Another hitch can be that an automatic (unsupervised) classification based only on the textual information leads to an ambiguous interpretation, in particular when the language is widely standardized. In these cases a supervised technique seems to be more powerful.

In a typical supervised learning scenario a training set is given, stated that some *a priori* knowledge on the document categories is necessary. The goal is to form a description that can be used to classify previously unseen elements. The performance of the classification task can be judged by trying out the rule that is learned on an independent set of data for which the true categories are known but hidden. The success rate on testing data gives an objective measure of how well the classification rule has been learned, even if in many practical mining applications it can be measured more subjectively in terms of how acceptable the learned description is to a human user.

Several studies have been conducted in the TC field by using different approaches, such as *K-Nearest Neighbour*, *Bayesian classifiers*, *Decision rules*, *Decision trees*, and *Support Vector Machines* (Manning and Schütze, 1999; Witten and Frank, 2005).

Association Rules (Agrawal et al., 1993) are an interesting way for dealing with text classification. A rule is a knowledge representation revealing the implicit relations among the documents collected in a given set. From a classification viewpoint, having selected a set of interesting rules, it is possible to classify all the transactions listed in the dataset. Many variations have been proposed, e.g. introducing additional information in the definition and extraction of rules. In some real cases association rules are not sufficient for satisfying a complex informative need, particularly when a final user is not deeply involved in the knowledge base development.

Decision Trees (DTs - Breiman et al., 1984) seem to meet better this special need. Typically a DT assigns each document to a given category taking into account the textual information, starting from a top node to a leaf node of the tree and classifying in this way the different documents in the same process. Differently from the association rules a graph representation that each user can intuitively read and interpret is obtained. Conversely DTs have very high computational cost because finding the "best" tree is an NP-hard problem. In such cases greedy heuristics are usually used.

Different empirical results obtained by using *Support Vector Machines* (SVMs - Vapnik, 1995; Burges, 1998) have confirmed a substantial improvement deriving by the use of such learning method generally in classification problems as well as in the specific domain of TC (Joachims, 1998; Sebastiani, 2002; Moschitti and Basili, 2004). SVM is based on the Structural Risk Minimization principle, whose goal is to find a hypothesis *h* such that the lowest true error can be guaranteed, where true error is defined as the probability that *h* will make an error on randomly selected examples. SVM avoids overfitting and achieves good generalization capability, especially in case of few instances, high dimensional feature spaces, and sparse data. Typically in a TC frame datasets have high dimensions (in particular when the language is not standardised) and each instance, which corresponds to a document, contains few non-null entries.

3.1. SVM-based Classification

SVM learns and applies an input-output model (*mapping*) with respect to a dataset. In a TC frame the input is a set of documents and the output is a set of categories. Given a *corpus*, we will refer to each document as an *instance* and to the different words as *attributes*. By pre-processing the *corpus* a subset of meaningful words is selected and a representation of the documents as vectors is obtained. Each element of a document/vector considers a measure of importance for the different words. Several schemes have been used in literature for expressing these importance (*e.g.*, Binary, Frequency, Tf-Idf, Hadamard representation).

Let $S = \{(\mathbf{x}_i, y_i)\ i = 1,..., N\}$ be the dataset consisting of N labelled instances $\mathbf{x}_i \in \Re^p$, where p is the number of attributes, and y_i the corresponding class label. The simplest case of classification considers a binary categorization and linearly separable classes.

SVM finds the *hyperplane* that separates the instances belonging to Class 1 from the ones of Class 2 with maximum margin (Fig. 2). The *margin* is defined as the distance between the separating hyperplane and the nearest instances. The found hyperplane is called *optimal separating hyperplane* (OSH).

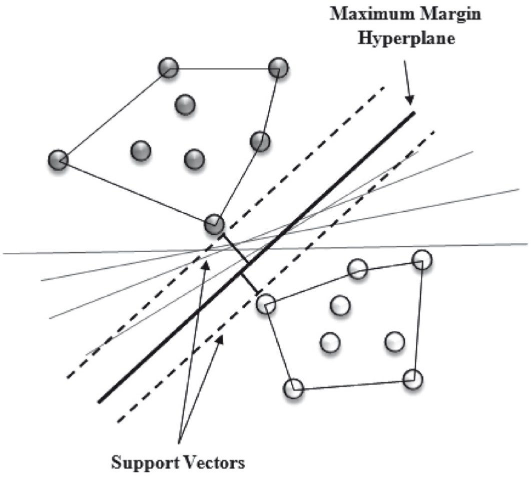

Figure 2: Linear SVM for two classes of objects

Let $f(\mathbf{x}) = (\mathbf{w}'\mathbf{x} + b)$ be the OSH, where **w** is the *norm vector* and *b* the *bias*. The points closest to the OSH, known as *support vectors*, are such that $|\mathbf{w}'\mathbf{x}_i + b| = 1$, whereas the other points are such that $|\mathbf{w}'\mathbf{x}_i + b| > 1$. Maximizing the margin means solving the following quadratic optimization problem:

$$\underset{\mathbf{w},b}{Min}\frac{1}{2}\|\mathbf{w}\|^2$$

$$y_i(\mathbf{w}'\mathbf{x}_i + b) \geq 1,\quad \forall i$$

The decision function used to predict the class of an unseen instance $\tilde{\mathbf{x}}$ is:

$$\tilde{y} = sgn[f(\tilde{\mathbf{x}})] = \begin{cases} +\text{üüü} \;\; f \;\; \tilde{\mathbf{x}} \;\; \geq \\ -\text{üüü} \;\; f \;\; \tilde{\mathbf{x}} \;\; < \end{cases}$$

Alternatively, for defining the OSH the dual form can be solved:

$$\underset{\text{á}}{Min}\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} y\,y_{\text{üüü}} \left(\mathbf{x} \;\cdot \mathbf{x} \;\right)\alpha\,\alpha \;-\sum_{i=1}^{N}\alpha$$

$$\sum_{i=1}^{N} y_i\alpha_i = 0,\;\; \alpha_i \geq 0,\;\; \forall i$$

where α_i is the *i-th* Lagrangian multiplier. Once the dual form has been solved and the Lagrangian multipliers have been determined, **w** and b can be easily derived in order to construct the decision function.

In some cases it does not exist a hyperplane that successfully separates the data points. To tackle this problem, the *slack variables* ξ_i (i=1,…,m) are introduced to relax the constraint $y_i(\mathbf{w}'\mathbf{x}_i + b) \geq 1$. If all the slack variables are null, then a linearly separable case is considered. The optimization problem for performing SVM on linearly non-separable data is known as *soft-margin SVM* :

$$\min\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \xi_i$$

$$y_i(\mathbf{w}'\mathbf{x}_i + b) \geq 1 - \xi_i,\;\; \forall i$$
$$\xi_i \geq 0,\;\; \forall i$$

where $C > 0$ is a parameter that trades off wide margin with a smaller number of margin failures.

Actually, most real problems involve non linearly separable data. SVM can be generalized to cases of non-linearly separable classes by mapping the data into a higher-dimensional space, where the OSH is defined. This embedding can be implicitly reached via a positive semi-definite function k known as *kernel* (Schölkopf and Smola, 2002) that measures the similarity between instances. Different kernel functions correspond to different embedding of data and consequently to different OSH. In this case, the OSH is found by solving the following quadratic programming problem:

$$\underset{\text{á}}{Min}\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} y\,y_{\text{üüü}}\; K\left(\mathbf{x} \;\cdot \mathbf{x} \;\right)\alpha\,\alpha \;-\sum_{i=1}^{N}\alpha$$

$$\sum_{i=1}^{N} y_i\alpha_i = 0,\;\; 0 \leq \alpha_i \leq C,\;\; \forall i$$

and the decision function is:

$$f(\mathbf{x}) = \sum_{i=1}^{N}\alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b$$

3.2. Kernel functions and textual data

In order to achieve good performance by using SVM it is crucial to select the most suitable kernel function. In Tab. 3 the most used kernel functions are reported.

Kernel Function	
Linear	$k(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i{}'\mathbf{x}_j$
Polynomial	$k(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i{}'\mathbf{x}_j + 1)^d,\ d \geq 0$
Gaussian (RBF)	$k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \left\Vert \mathbf{x}_i - \mathbf{x}_j \right\Vert^2)$
Sigmoid	$k(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\gamma \mathbf{x}_i{}'\mathbf{x}_j + r)$

Table 3: Common kernel functions

Since the performance (i.e., the generalization capability of SVM) is strictly affected by the choice of the kernel function and there is no theory concerning how to choose a suitable kernel, such a choice and parameters tuning are usually carried out by using a grid-search approach. The relative performance is evaluated by *cross-validation*.

In the specific domain of TC some peculiar kernels have been proposed by considering different criteria, like the *string subsequence kernel* (Lodhi et al., 2003), based on the similarity between documents in terms of substring in common, and the *semantic kernel* (Basili et al., 2005) that evaluates semantic similarity between documents by considering some external linguistic resources like WordNet.

This kernel seems to be more effective in an IR frame, because the classification task aims at organizing the knowledge base in view of searching the most relevant information. In Digital Protocol a classification is mainly needed for maintenance purposes, because each kind of document is received or used by specific employers, defining *a priori* some routing rules. For this reason in the following only the common kernels have been evaluated, trying to optimize the accuracy by tuning the different parameters.

4. Experimental set-up, evaluation and validation

In order to evaluate the various choices in SVM, and find the best classifier, two different representations of the data have been used, by examining the original frequencies of the lexical table (number of occurrences of each word in each document) and a binary transformation of the data (presence/absence of each word in each document). A feature selection has been implicitly obtained in § 2.2 by performing CA on the lexical table, so that the first 50, 100, 200 and 400 factorial coordinates obtained in such way will be used in the following (51.0%, 71.3%, 91.0%, 99.9% of total inertia, respectively). The Tf-Idf scheme, widely used in TC, has not been considered in our strategy because of the scaling effect on the uncommon words. The peculiar nature of the analysed documents, where a standardised bureaucratic language is used, leads to consider useful in the training step both common and rare words.

Because of the small dimension of the dataset, a 10‰10-fold cross-validation has been carried out. The dataset has been randomly split into 10 parts, training in each step the classifiers on 9 parts (387 documents) and reserving a part as testing set (43 documents), per 10 times. The resampling ensures that the results among the different 100 runs are more robust, and that the performance comparisons are reliable.

SVM has been performed using libSVM (Chang and Lin, 2001). Having a multi-class problem, with 8 categories, a *one-against-one* approach has been considered by training 28 binary models and combining the results into one single prediction (Hsu and Lin, 2002). For better evaluating the different SVM-based classifiers, other unsupervised (*Kmeans*) and supervised (*C4.5*, *Random Tree*) methods have been also tested.

The performances have been compared in terms of *mean accuracy* (mean of the accuracies across class) as it is shown in Tab. 4.

Dataset	*Kmeans*	*C 4.5*	*Random Tree*	*SVM (linear)*	*SVM (poly d=2)*	*SVM (poly d=3)*	*SVM (gaussian)*	*SVM (sigmoid)*
CA_50	55.4 ± 8.7	81.6 ±4.6	79.1 ±5.4	84.5 ±3.8	86.2 ±4.4	87.1 ±4.6	86.3 ±4.0	82.6 ±3.4
CA_100	57.3 ±10.3	81.8 ±4.9	77.2 ±6.3	86.3 ±4.4	85.7 ±4.2	85.9 ±4.3	84.7 ±3.9	83.7 ±3.0
CA_200	57.2 ±13.6	82.0 ±4.8	72.0 ±6.7	88.4 ±4.0	86.7 ±4.0	86.3 ±4.1	82.1 ±4.0	84.2 ±3.1
CA_400	62.3 ±12.2	80.7 ±4.9	65.6 ±8.0	88.0 ±4.1	87.2 ±4.1	87.0 ±4.0	80.6 ±4.3	84.3 ±3.2
Binary	51.4 ±10.3	84.9 ±4.6	77.1 ±5.5	**89.8 ±4.1**	87.8 ±3.8	86.6 ±4.1	59.5 ±1.1	65.1 ±3.1

Tab. 4: Mean accuracy (in %) and standard deviation on 10x10 CV

A 95% paired t-test has been performed on the different mean accuracies. This test of significance is necessary for confirming that a higher performance only depends on the generalising capability of the classifier, and not on the natural variability of training and testing set. A first result of the t-test is that the performances of SVM classifiers are significantly better than the other methods. Focusing deeply on the different SVMs, a better performance has been obtained on linear SVM with a binary dataset (89.8%). In the other cases, even if the accuracy is higher in some SVM with respect to others (e.g., on CA_200 a mean accuracy of 88.4% vs. 86.7% has been obtained in linear SVM and polynomial SVM with d=2, respectively), the t-tests have shown that the differences are not statistically significant.

A not surprising result is that a simple transformation of the data can be more effective with respect to other complex transformations. The feature selections carried out with CA has shown that the accuracy does not increase. This evidence has been noticed also in other studies (Manevitz and Yousef, 2001). A deep investigation is however necessary, in order to correctly validate these results. Moreover, even if the number of runs in cross validation is quite high, it will be clearly necessary to train the classifiers on huger dataset and more categories.

Precision, Recall and F1-measure for each category, with respect to the best classifier, are detailed in Tab. 5. It is possible to note that some cases are peculiar, because of their specific nature. A special case seems to be the category *Fin. e contabilità*, with null values on all measures. This means that the semantic content of this kind of documents is not clearly identifiable. It is also interesting to point out that the misclassified documents have been included in the category *Direttore*, which usually considers different kind of documents (*e.g*., official letters, memoranda, administrative questions). A higher number of documents belonging to this category would be probably necessary for better training the classifier.

Category	*Precision*	*Recall*	*F1-measure*
Acquisti	0.85	0.80	0.82
Direttore	0.91	0.96	0.94
Fatture	0.90	0.91	0.91
Fin. e contabilità	0.00	0.00	0.00
Iva	1.00	0.75	0.86
Mandati	1.00	1.00	1.00
Resp. progetto	0.50	0.40	0.44
Personale	0.20	0.11	0.14

Tab. 5: Precision, Recall and F1-measure per category on 10x10 CV

Even if the obtained results are encouraging, in a real implementation of an automatic procedure for Digital Protocol other aspects should be considered. Because of the legal nature of the recording and preserving steps, a percentage of correctly classified documents sufficiently high in other domains it is not appropriate in an official context. An effort for further reducing the error rate has to be carried out.

References

Agrawal R., Imielinski T. and Swami A. (1993). Mining Association Rules between Sets of Items in Large Dataset. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pp. 207-216.

Balbi S. and Misuraca M. (2005). Pesi e Metriche nell'Analisi dei Dati Testuali. *Quaderni di Statistica*, vol. 7: 55-68.

Basili R., Cammisa M. and Moschitti A. (2005). Effective use of Wordnet semantics via kernel based learning. In *Proceedings of the 9th Conference on Computational Natural Language Learning*, pp. 1-8.

Breiman L., Friedman J.H., Olshen R.A. and Stone C.J. (1984). *Classification and Regression Trees*. Belmont (Calif): Wadsworth.

Burges C.J.C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, vol. 2: 121-167.

Chang C.C. and Lin C.J. (2001). *LIBSVM: A library for support vector machines* (software available at: http://www.csie.ntu.edu.tw/~cjlin/libsvm).

Hsu C.W. and Lin C.J. (2002). A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, vol. 13(2): 415-425.

Joachims T. (1998). Text Categorization with Support Vector Machine: learning with many relevant features. In *Proceedings of the European Conference on Machine Learning*, pp. 137-142.

Lodhi H., Saunders C., Shawe-Taylor J., Cristianini N. and Watkins C. (2002). Text Classification using String Kernels. *Journal of Machine Learning Research*, vol. 2: 419-444.

Manevitz L. and Yousef M. (2001). One-class document classification via Neural Networks. *Neurocomputing*, vol. 70: 1466-1488.

Manning C.D. and Schütze H. (1999). *Foundations of Statistical Natural Language Processing*. Cambridge (MA): The MIT Press.

Moschitti A. and Basili R. (2004). Complex Linguistic features for text classification: a comprehensive study. In *Proceedings of the European Conference on Information Retrieval*, pp. 181-196.

Schölkopf B. and Smola A. (2002). *Learning with kernels - Support vector machines, regularization, optimization and beyond*. Cambridge (MA): The MIT Press.

Sebastiani F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, vol. 34 (1): 1-47.

Vapnik V. (1995). *The nature of statistical learning theory*. New York: Springer.

Witten I.H. and Frank E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. San Francisco: Elsevier.