INTERNATIONAL JOURNAL OF INTELLIGENT CONTROL AND SYSTEMS
VOL.12, NO.2 JUNE 2007 150-157

Using Taxonomic Domain Knowledge in Text Categorization Tasks

Giuliano ARMANO, Francesco MASCIA, and Eloisa VARGIU

Abstract— **In this paper we present a "progressive filtering" technique aimed at improving the performances of a multiagent system devised to perform text categorization. The technique exploits the discriminant capabilities of multiple classifiers organized into a taxonomy and is aimed at coping with a problem that occurs very often in text categorization tasks, i.e. with the unbalance –for any category– between relevant and non relevant inputs. Experiments, performed on the RCV1-v2 benchmark, highlight the validity of the approach.**

Index Terms— **Agent-Based Systems, text categorization.**

1. INTRODUCTION

Text categorization can be defined as the task of determining and assigning topical labels to content. The more the amount of available data (e.g., in digital libraries), the greater the need for high-performance text categorization algorithms.

In the literature, many machine learning approaches have been proposed, including regression models [19], [48], nearest neighbor classification [27], [33], [49], [50], Bayes probabilistic approaches [5], [29], [32], [35], [44], [53], decision trees [3], [19], [29], inductive rule learning [2], [9], [10], [34], artificial neural networks (ANNs) [47], online learning [10], [30], and support vector machines (SVMs) [23].

Text categorization has received great attention also from the community that investigates ensembles of classifiers. In fact, there is strong theoretical and experimental evidence that combining multiple classifiers can actually boost the performance over a single classifier. Let us recall, here, (i) the work of Sebastiani [41] and of Schapire [39], which fall in the general category of boosting [40]), (ii) the work of Larkey [28] and Yang et al. [52], aimed at assessing the impact of output combination techniques, and (iii) the work of Dong [15], where several input subsampling techniques (with heterogeneous classifiers) are experimented.

Furthermore, in the last years several researchers have investigated the use of hierarchies for text categorization (see, for instance, [8], [16]), which is also the main focus of our proposal. In fact, in this paper, we study the impact of domain knowledge provided in form of taxonomy over the capability to discriminate between relevant and non-relevant items, in particular taking into account the problems that originate from an unbalance between the formers and the latters. To this end, we tackle the problem of text categorization by resorting to multiple classifiers derived from a suitable taxonomy able to represent topics deemed relevant to the domain being investigated. In particular, each item to be classified undergoes progressive filtering by the pipelines of classifiers that originate from the adopted taxonomy. To assess the capabilities of a technique based on progressive filtering, we devised a multiagent system specifically tailored for a relevant task, i.e. news categorization. Tests have been performed on the RCV1-v2 [31] standard document collection.

The remainder of the paper is organized as follows: In section 2 some relevant work is briefly recalled. Section 3 illustrates the multiagent system from functional and architectural perspectives. Section 4 is focused on the proposed progressive filtering technique. Section 5 reports and discusses experimental results. In Section 6 conclusions are drawn and future work is pointed out.

G. Armano, F. Mascia, and E. Vargiu are with the Department of Electrical and Electronic Engineering, University of Cagliari.

2. RELATED WORK

Existing document collections suffer from one or more of the following drawbacks: (i) few documents, (ii) lack of the document full text, (iii) inconsistent or incomplete category assignments, (iv) peculiar textual properties, and (v) limited availability. Furthermore, typically, researchers do not have documentation on how collections were produced, and on the nature of the underlying categories. These problems are particularly severe in hierarchical text categorization, where researchers often impose their own hierarchies. In particular, several proposals of hierarchical methods for text classification have been made using the Reuters standard document collection, along with the definition of suitable class hierarchies.

In [24] the Reuters-22173 collection has been adopted to perform experiments. First, a small hierarchical subset of Reuters-22173 has been generated by identifying labels that subsume other labels. Then, experiments have been performed comparing a classifier based on Naive Bayes with two limited-dependency Bayes net classifiers –both on flat and hierarchical models. Documents are classified into the hierarchy by first filtering them through the single best-matching first-level class and then sending them to the appropriate second level. This approach showed that hierarchical models perform well when a very small number of features per class (about 10) is used. No advantages were found using the hierarchical model for larger numbers of features.

Reuters-22173 collection has also been used in [46]. An exploratory cluster analysis was used to extract an implicit hierarchical structure, subsequently validated by an expert of the domain. The proposed architecture matches the hierarchical structure of the topic space, as opposed to a flat model that ignores the structure. An architecture based on ANNs has been adopted and several input representations taken into account. Information from each level of the hierarchy has been

combined in a multiplicative fashion, so that no hard decisions have to be made except at the leaf nodes. On the average, an improvement of 5% in precision has been obtained for the hierarchical representation with respect to the flat one.

In [14] the Reuters-21578 collection has been used, together with the adoption of a hierarchy based on the one proposed in [21]. Compared with the corresponding flat model, the proposed hierarchical model showed an improvement of about 2-4% in precision and recall.

A hierarchical version of the Reuters collection has been adopted also in [36], but no comparisons have been provided between the proposed hierarchical approach and the corresponding flat model.

3. A Multi-Agent System for Information Extraction and Text Categorization

From a functional point of view, the system implemented to assess the capabilities of the proposed progressive filtering technique is organized into three layers, entrusted with performing the following activities: (i) extracting the required information from web sources, (ii) categorizing items according to a given taxonomy, and (iii) providing suitable feedback mechanisms. In this paper, we will be mainly concerned with the text categorization task, although the overall system will be sketched to give the reader a better understanding of the underlying application task.

From an architectural point of view, the system has been built upon PACMAS (Personalized Adaptive and Cooperative MultiAgent System), a generic multiagent architecture aimed at retrieving, filtering, and reorganizing information according to the users' interests [4]. The adoption of PACMAS is motivated by the willing of better concentrating on the above aspects separately, as it is in fact a layered architecture capable of promoting the decoupling among all relevant aspects of a complex task aimed at performing information retrieval.

3.1. Functional View

The functional view of the system that has been devised to perform the selected task is sketched in Figure 1.

Information Extraction: The information extraction activity is essential to retrieve documents provided by heterogeneous and distributed sources, such as web sites, digital archives, and web services. In the literature, several tools have been proposed to better address the issue of generating wrappers for web data extraction [26]. Such tools are based on distinct techniques, such as declarative languages [12], [20], HTML structure analysis [13], [38], natural language processing [18], [43], machine learning [22], [25], data modeling [1], [7], and ontologies [17].

Text Categorization: Currently, the proposed system is aimed at progressively filtering news that flow from heterogeneous sources to the end user. First, documents are classified according to a high-level taxonomy, which in principle should be independent from the specific user. For the sake of testing, we adopted the RCV1-taxonomy (Figure 2 reports part of the branch corresponding to the *economics* topic).

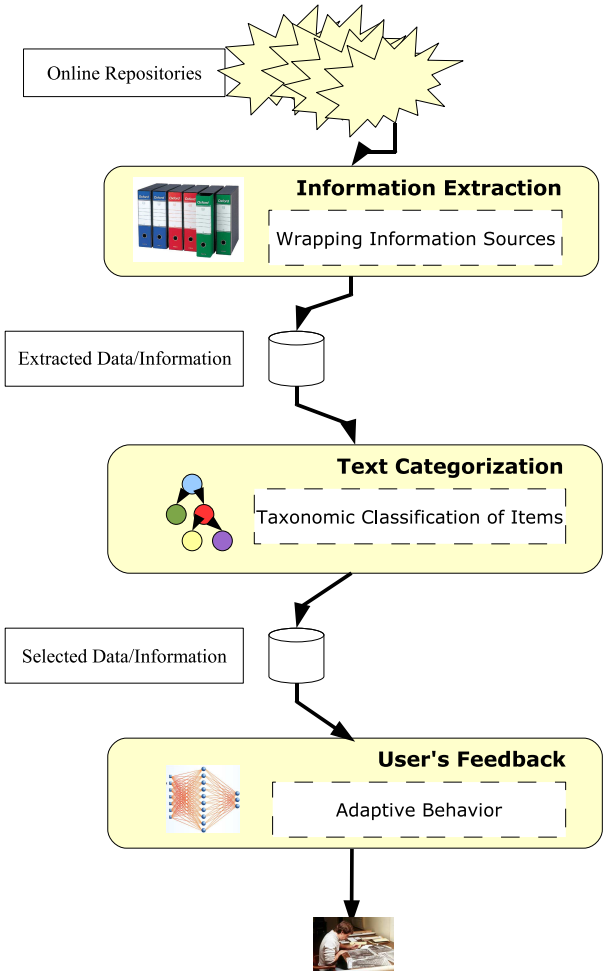

Fig. 1. The functional view for information extraction and text categorization

Given a taxonomy, in our view there are two ways of combining classifiers to enforce progressive filtering techniques: "horizontal" and "vertical". The former occurs in accordance with the typical linguistic interpretation of the logical connectives "and", "or", and "not", whereas the latter exploits the ability of a pipeline of classifiers to progressively filter out non relevant information. The analysis of "horizontal" combination being out of the scope of this paper, let us concentrate on the latter.

Let us point out in advance that particular care has been taken in using pipelines to limit the phenomenon of false negatives (FN), as a user may accept to be signaled about an article which is actually not relevant, but –on the other hand– would be disappointed in the event that the system disregards an input which is actually relevant. This behavior can be imposed in different ways, the simplest being lowering the threshold used to decide whether an input is relevant or not. In a typical text categorization system this operation may have a negative impact on false positives (FP), i.e. augmenting their presence. The adoption of the proposed text categorization technique allows to reduce this unwanted effect, thanks to the progressive-filtering ability exhibited by the hierarchy of classifiers defined in accordance with the corresponding taxonomic knowledge.

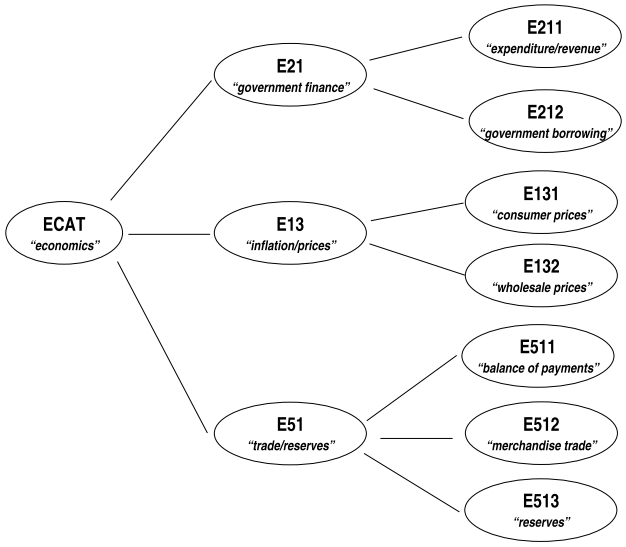

Fig. 2. A portion of the RCV1-taxonomy.

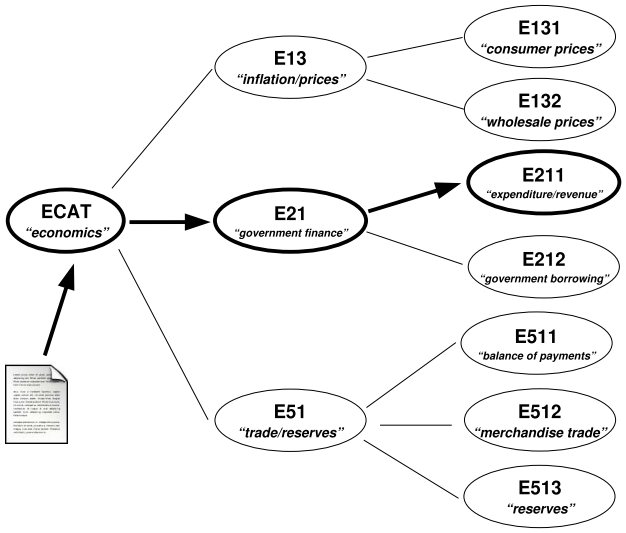

Fig. 3. An example of pipeline.

To illustrate the underlying mechanism, let us consider the taxonomy reported in Figure 2, where each node represents a classifier entrusted with recognizing all corresponding relevant inputs. Any given input traverses the taxonomy as a "token", starting from the root (ECAT). If the current classifier recognizes the token as relevant, it passes it on to all its children (if any), and so on. The typical result consists of activating one or more pipelines within the taxonomy. Figure 3 illustrates the pipeline activated by an input document, which encompasses the categories economics (ECAT), government finance (E21), and expenditure/revenue (E211). This means that all involved classifiers recognize the input as relevant.

Furthermore, we expect that most articles are non relevant to the user, the ratio between negative and positive examples being high (typical orders of magnitude are 10^2 - 10^3). Unfortunately, this aspect has a very negative impact on the precision of the system. On the other hand, combining classifiers should allow to reduce this negative effect –in the best case exponentially with respect to the number of classifiers that occur in the combination. Our experimental results confirm this hypothesis, although the actual impact of resorting to pipelines of classifiers is not as high as the theoretical one, due to the existing correlation between the classifiers actually involved in the combination.

User's Feedback: So far, a simple solution based on the *k*-NN technology has been implemented and experimented to deal with the problem of supporting the user's feedback. When a non-relevant article is evidenced by the user, it is immediately embedded in the training set of the *k*-NN classifier that implements the user feedback. A suitable check performed on this training set after inserting the negative example allows to trigger a procedure entrusted with keeping the number of negative and positive examples balanced. In particular, when the ratio between negative and positive examples exceeds a given threshold (by default set to 1.1), some examples are randomly extracted from the set of "true" positive examples and embedded in the above training set.

3.2. Architectural View

As already pointed out, the functional view described in the previous section has been implemented by suitably customizing the generic PACMAS architecture [4]. This section briefly recalls the main customizations performed on PACMAS to give rise to the system devised to perform text categorization.

The PACMAS generic architecture has been implemented using JADE [6] as underlying agent-based infrastructure. PACMAS, depicted in Figure 4, encompasses four main levels: information, filter, task, and interface. Each level is associated with a specific role, and the communication between adjacent levels is achieved through suitable middle agents –which form corresponding mid-span levels.

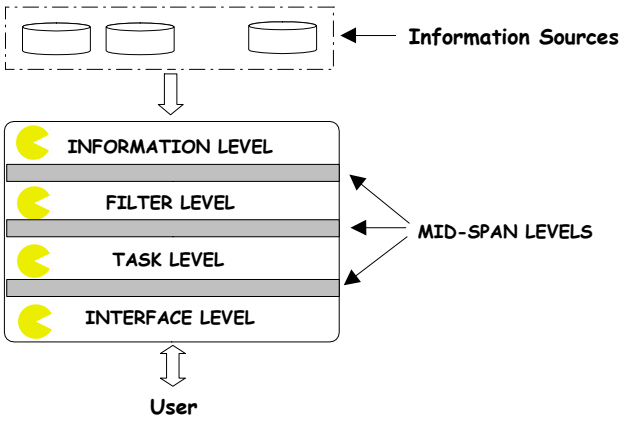

Fig. 4. The PACMAS generic architecture.

Information Level: At the information level, agents are entrusted with extracting data from the information sources. Each information agent is associated with one information source, playing the role of a wrapper. In particular, a wrapper able to deal with the *RSS* (Really Simple Syndication) format and a wrapper able to embed the Reuters document collection have been implemented so far. Furthermore, a suitable wrapper agent has been devised to embed the adopted taxonomy.

Filter Level: At the filter level, agents are aimed at suitably encoding of the text content to facilitate the work of agents belonging to the task level. To this end, all non-informative words such as prepositions, conjunctions, pronouns and very common verbs are removed using a stop-word list. After that, a standard stemming algorithm [37] removes the most common morphological and inflexional suffixes. Then, for each category of the taxonomy, feature selection, based on the information-gain heuristics, has been adopted to reduce the dimensionality of the feature space. Furthermore, filter agents cooperate to prevent information from being overloaded and redundant.

Task Level: At the task level, agents arrange data according to users' personal needs and preferences. In a sense, they can be considered as the core of the architecture. In fact, they are devoted to achieve users' goals by cooperating together and adapting themselves to the changes of the underlying environment. In particular, a classifier for each item in the taxonomy has been implemented by resorting to the *k*-NN technique [51], in its "weighted" variant [11]. The motivation for the adoption of this particular technique stems from the fact that it does not require specific training and is very robust with respect to noisy data. Task agents are trained in order to recognize a specific category, each category being an item of the adopted taxonomy. Given a document in the test set, each agent, through its embedded *k*-NN classifier, ranks its nearest neighbors among the training documents according to an Euclidean distance measure, and uses the most frequent category of the *k* top-ranking neighbors to predict the categories of the input document.

Interface Level: At the interface level, agents and users interact through a suitable graphical interface –an agent being associated with each different user interface. In fact, a user can generally interact with an application through several interfaces and devices (e.g., PCs, PDAs, MIDP devices). Interface agents are also devoted to handle user profile and to propagate it by the intervention of middle agents.

Mid-span Level: At mid-span levels, agents are aimed at establishing communication among requesters and providers belonging to two adjacent main levels. In particular, middle agents take care of forwarding relevant information back and forth throughout the main levels. For instance, when filter and task agents need information contained in the user profile (which is stored at the interface level), they ask the corresponding middle agent to retrieve it.

4. Assessing the Impact of Progressive Filtering Techniques for Text Categorization

Let us point out that a main issue in news categorization is how to deal, for each category, with an unbalance between relevant and non-relevant items. With the aim of assessing how much the use of pipelines of classifiers allows to cope with this phenomenon, let us assume that the normalized confusion matrix associated with a classifier is denoted as:

$$\begin{vmatrix} c_{00} & c_{01} \\ c_{10} & c_{11} \end{vmatrix}$$

where the first and the second index represents the actual relevance of the input and the way it has been classified. For instance, c_{01} represents (an estimation of) the probability to classify as relevant (1) an input that is in fact non relevant (0). If the set of inputs submitted to a classifier contains n_0 non relevant inputs and n_1 relevant inputs, the (expected) confusion matrix that characterizes the process is:

$$\begin{vmatrix} n_0 \cdot c_{00} & n_0 \cdot c_{01} \\ n_1 \cdot c_{10} & n_1 \cdot c_{11} \end{vmatrix}$$

Hence:

$$P = \frac{TP}{TP+FP} = \left(1+\frac{FP}{TP}\right)^{-1} = \left(1+\frac{c_{01}}{c_{11}}\cdot\frac{n_0}{n_1}\right)^{-1} \quad (1)$$

$$R = \frac{TP}{TP+FN} = \left(1+\frac{FN}{TP}\right)^{-1} = \left(1+\frac{c_{10}}{c_{11}}\right)^{-1} \quad (2)$$

To study how a taxonomy of classifiers affects the overall capability of classifying inputs, some simplifying assumptions have been made, which actually do not properly represent the real world but help to understand the basic underlying mechanisms. In particular, having to deal with a pipeline of *k* classifiers linked by an *is-a* relationship, for the sake of simplicity let us assume that (i) each classifier in the pipeline has the same (normalized) confusion matrix and that (ii) classifiers are in fact independent. Under these simplifying assumptions, a classification can be seen as a pseudo-random process, which takes as input a set of article descriptions and outputs their classification, which necessarily fulfills, on the average, the requirements imposed by the corresponding confusion matrix.

It can be easily verified that the effect of using a pipeline of *k* classifiers on precision and recall is:

$$\begin{aligned} P^{(k)} &= \frac{TP^{(k)}}{TP^{(k)}+FP^{(k)}} = \\ &= \left(1+\frac{FP^{(k)}}{TP^{(k)}}\right)^{-1} = \left(1+\frac{c_{01}^k}{c_{11}^k}\cdot\frac{n_0}{n_1}\right)^{-1} \end{aligned} \quad (3)$$

$$\begin{aligned} R^{(k)} &= \frac{TP^{(k)}}{TP^{(k)}+FN^{(k)}} = \\ &= \left(1+\frac{FN^{(k)}}{TP^{(k)}}\right)^{-1} = \left(1+\frac{c_{10}}{1-c_{11}}\cdot\frac{1-c_{11}^k}{c_{11}^k}\right)^{-1} \end{aligned} \quad (4)$$

The equations above show that an unbalance of positive and negative examples (which is the typical case in text categorization problems) can be suitably dealt with by keeping FN (i.e., c_{10}) low and by exploiting the filtering effect of classifiers in the pipeline. The former aspect affects the recall, whereas the latter allows to augment the precision according to the number of levels of the given taxonomy. As already pointed out, the above relations have been obtained by making simplifying assumptions. Nevertheless, our preliminary results show that

their validity is maintained also in practice, provided that a relatively low degree of correlation holds among classifiers in pipeline. As a final comment on "vertical" combination, we decided that the strength assigned by a pipeline of classifiers to a relevant input (i.e. deemed relevant by all classifiers in the pipeline) is the minimum value in [0,1] received by the input along the pipeline.

5. Experimental Results

To assess the capabilities of the proposed progressive filtering technique, different kinds of tests have been performed, each aimed at highlighting –and getting information about– a specific issue. In particular, we assessed the impact of taking into account pipelines of classifiers, also trying to assess whether a residual independence was in fact present.

To this end, we resorted to different metrics aimed at evaluating precision *P* and recall *R*. In particular, we used micro- and macro-averaging, together with two evaluation metrics (i.e., F_1 and the *PR* curve) obtained by moving the acceptation threshold of the classifier(s) under investigation over the range $[0, 1]$. A concise recall of the corresponding definitions follows (the interested reader may consult the corresponding literature, e.g. [42]).

As for micro- and macro-averaging, they are aimed at obtaining estimates of *P* and *R* relative to the whole category set. In particular, micro-averaging evaluates the overall *P* and *R* by globally summing over all individual decisions. In symbols:

$$P^{\mu} = \frac{TP}{TP+FP} \tag{5}$$

$$R^{\mu} = \frac{TP}{TP+FN} = \frac{\sum_{i=1}^{m} TP_i}{\sum_{i=1}^{m}(TP_i+FN_i)} \tag{6}$$

where the "μ" superscript stands for microaveraging. On the other hand, macro-averaging first evaluates *P* and *R* "locally" for each category, and then "globally" by averaging over the results of the different categories. In symbols:

$$P^{M} = \frac{\sum_{i=1} mP_i}{m} \tag{7}$$

$$R^{M} = \frac{\sum_{i=1} mP_i}{m} \tag{8}$$

where the "M" superscript stands for macroaveraging.

As for F_1 [45], it is obtained from a more general definition by imposing that *P* and *R* have the same degree of importance. In symbols:

$$F_1 = \frac{2PR}{P+R} \tag{9}$$

The PR curve [45], given in terms of *P* and *R* is a variant of the ROC curve. The ROC curve is a graphical plot of the sensitivity vs. (1 - specificity) for a binary classifier system as its discrimination threshold is varied, and gives information about the discrimination capability of a classifier with respect to a given set of inputs.

If the selected test set is statistically significant, the ROC (PR) curve actually provides information about the overall discrimination ability of a classifier and / or on the separability of the input space by means of the features selected for representing inputs. Since the given domain is typically affected by noise (i.e., it is very difficult to come up with a description able to enforce a good separation between relevant and non relevant inputs), moving the decision threshold in either direction typically affects both FN and FP. In particular, the typical expected behavior is that an attempt of lowering FN produces the effect of augmenting FP.

Tests have been performed using RCV1-v2, the standard document collection proposed in [31]. First, for each document, the most discriminant 200 features ($TFIDF$) have been selected according to the information gain method. Then, for each node of the taxonomy, a learning set of 500 articles, with a balanced set of positive and negative examples, has been selected to train a classifier based on the *wk*-NN technology. As for the test, different randomly selected sets of 1000 documents have been generated –characterized by a different ratio between relevant and non-relevant inputs. In particular, the percent of positive examples has been set to 50%, 10%, 5%, and 1% (say $TS_{50}, TS_{10}, TS_5, and TS_1$, respectively).

To study the impact of progressively filtering information with pipelines of *wk*-NN classifiers (denoted as PIPE), we tested with the above test sets some relevant pipelines, each concerning three nodes of the taxonomy. Results have been compared with those obtained by running the same tests on three classifiers based on the following technologies: *wk*-NN (denoted as WKNN), linear SVM (denoted as SVM[1]), and RBF-SVM (denoted as SVM[2]). [1] As shown in Table I, in all selected samples, the distributed solution based on multiple classifiers has reported better results than those obtained with flat models.

As for P and R, let us consider –for the sake of brevity– only one specific pipeline (taking into account, anyway, that it is representative of a typical behavioral pattern). Figure 5 reports the curves F1 and PR, obtained by running on the selected tests: the pipeline (PIPE) corresponding to Figure 3, as well as the classifiers (WKNN, SVM[1], and SVM[2]) trained to recognize as relevant only inputs belonging to the E211 category (which coincides the "bottom" of the pipeline). Experimental results show that the filtering effect of a pipeline is not negligible, although not as strong as the one emphasized by theoretical results –due to the correlation among classifiers involved in a pipeline. In particular, in presence of unbalanced inputs, a pipeline of three classifiers is able to counteract an unbalance of up to 100 non relevant articles vs. one relevant article.

6. Conclusions and Future Work

In this paper a multiagent system designed to perform text categorization has been presented, focusing on its capability of progressively filtering information by resorting to a suitable taxonomy. Each node of the taxonomy gives rise to a specific classifier, and is linked to other classifiers according to the given *is-a* relation. The system has been built by customizing

[1]The technique based on *wk*-NN has been used both with the hierarchical classification (PIPE) and with the flat model (WKNN).

TABLE I

MICRO- AND MACRO-AVERAGING.

pos	f_1^{μ}WKNN	f_1^{M}WKNN	f_1^{μ}SVM^1	f_1^{M}SVM^1	f_1^{μ}SVM^2	f_1^{M}SVM^2	f_1^{μ}Pipe	f_1^{M}Pipe
50	0,883	0,883	0.831	0.832	0,898	0,897	0.905	0.905
10	0,646	0,647	0.507	0.521	0,719	0,722	0.721	0.720
5	0,513	0,514	0.412	0.428	0,535	0,543	0.683	0.682
1	0,165	0,169	0.169	0.190	0,344	0,349	0.412	0.431

PACMAS, a generic architecture for implementing Personalized, Adaptive, and Cooperative MultiAgent Systems for information retrieval. Experiments have been conducted on the standard RCV1-v2 Reuter's collection. Since –at least theoretically– the filtering activity goes with the power of the number of classifiers involved in the pipeline, it is easy to verify that the process of progressively filtering articles could also counteract a ratio between non relevant and relevant articles with an order of magnitude of hundreds or thousands, provided that the number of levels of the underlying taxonomy is deep enough.

As for the future work, we are implementing a new release of the system in which the core of text categorization is performed by alternative algorithms (including Support Vector Machines).

REFERENCES

[1] B. Adelberg, "NoDoSEa tool for semi-automatically extracting structured and semistructured data from text documents", in *Proceedings of the 1998 ACM SIGMOD international Conference on Management of Data* (Seattle, Washington, United States, June 01 - 04, 1998). A. Tiwary and M. Franklin, Eds. SIGMOD '98. ACM Press, New York, NY, 1998, pp. 283-294.

[2] C. Apte, F. Damerau, and S.M. Weiss. "Towards language independent automated learning of text categorization models", in *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1994, pp. 23–30.

[3] C. Apte, F. Damerau, and S.M. Weiss, "Text mining with decision rules and decision trees", in *Proceedings of Workshop Learning from text and the Web, co-located with the 1998 Conference on Automated Learning and Discovery*, Pittsburgh, PA, 1998, pp. 487–499.

[4] G. Armano, G. Cherchi, A. Manconi, and E. Vargiu, "PACMAS: A personalized, Adaptive, and Cooperative MultiAgent System Architecture", in *Workshop dagli Oggetti agli Agenti, Simulazione e Analisi Formale di Sistemi Complessi (WOA 2005)*, 2005, pp. 54–60.

[5] L. D. Baker and A. K. Mccallum, "Distributional clustering of words for text categorization", in *Proceedings of the 21th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1998, pp. 96–103.

[6] F.L. Bellifemine, G. Caire, and D. Greenwood, *Developing Multi-Agent Systems with JADE (Wiley Series in Agent Technology)*, John Wiley and Sons, 2007.

[7] B. Ribeiro-Neto, A.H. Laender, and A.S. da Silva, "Extracting semi-structured data through examples", in *Proceedings of the Eighth international Conference on information and Knowledge Management* (Kansas City, Missouri, United States, November 02 - 06, 1999), S. Gauch, Ed. CIKM '99. ACM Press, New York, NY, 1999, pp. 94–101.

[8] L. Cai and T. Hofmann, "Hierarchical document categorization with support vector machines", in *Proceedings of the 13th ACM international Conference on information and Knowledge Management* (Washington, D.C., USA, November 08 - 13, 2004), ACM Press, New York, NY, 2004, pp. 78–87.

[9] W.W. Cohen, "Text categorization and relational learning", in *Proceedings of the 12th International Conference in Machine Learning*, 1995, pp.124–132.

[10] W.W. Cohen and Y. Singer, "Context-Sensitive Learning Methods for Text Categorization", in *ACM Transactions on Information Systems*, 17(2), (Apr. 1999), 1999, pp. 141–173.

[11] W. Cost, S. Salzberg, "A weighted Nearest Neighbor Algorithm for Learning with Symbolic Features", *Machine Learning*, Vol. 10, 1993, pp. 57–78.

[12] V. Crescenzi and G. Mecca, "Grammars Have Exceptions", *Information Systems*, Vol. 23 (8), 1998, pp. 539–565.

[13] V. Crescenzi, G. Mecca, and P. Merialdo, "Roadrunner, Towards Automatic Data Extraction from Large Web Sites", in *Proceedings of the 27th International Conference on Very Large Data Bases*, 2001, pp. 109–118.

[14] S. D'Alessio, K. Murray, R. Schiaffino, and A. Kershenbaum, "Category levels in hierarchical text categorization", in *Proceedings of the of the 3rd Conference on Empirical Methods in Natural Language Processing*, Granada, Spain, Association for Computational Linguistics, Morristown, 1998.

[15] Yan-Shi Dong, Ke-Song Han, "A Comparison of Several Ensemble Methods for Text Categorization", 2004 IEEE International Conference on (SCC'04), pp. 419-422.

[16] S. Dumais, and H. Chen, "Hierarchical classification of Web content", in *Proceedings of the 23rd Annual international ACM SIGIR Conference on Research and Development in information Retrieval* (Athens, Greece, July 24 - 28, 2000), ACM Press, New York, NY, 2000, pp. 256–263.

[17] D. W. Embley, D. M. Campbell, Y. S. Jiang, S. W. Liddle, Y. K. Ng, D. Quass, and R. D. Smith, "Conceptual-Model-Based Data Extraction from Multiple-Record Web Pages", in *Data Knowledge Engineering*, Vol. 31(3), 1999, pp. 227–251.

[18] D. Freitag, "Machine Learning for Information Extraction in Informal Domains", Ph.D. dissertation, Carnegie Mellon University, 1998.

[19] N. Fuhr, S. Hartmann, G. Knorz, G. Lustig, M. Schwantner, and K. Tzeras, "AIR/X – a Rule-Based Multistage Indexing System for Large Subject Fields", in *Proceedings of RIA0-91, the 3rd International Conference "Recherches d'Information Assistee par Ordinateur"*, Elsevier Science Publishers, Amsterdam, NL, Barcelona, ES, 1991, pp. 606–623.

[20] J. Hammer, H. Garcia-Molina, S. Nestorov, R. Yerneni, M. Breunig, and V. Vassalos, "Template-Based Wrappers in the TSIMMIS system", in *Proceedings of the 1997 ACM SIGMOD international Conference on Management of Data* (Tucson, Arizona, United States, May 11 - 15, 1997), J. M. Peckman, S. Ram, and M. Franklin, Eds. SIGMOD '97. ACM Press, New York, NY, 1997, pp. 532–535.

[21] P. Hayes and S. Weinstein, "CONSTRUE/TIS: A System for Content-Based Indexing of a Database of News Stories", in *Proceedings of the 2nd Annual Conference on Innovative Applications of Artificial Intelligence*, AAAI Press/MIT Press, Cambridge, Mass., 1991, pp. 49–64

[22] C. N. Hsu and M. T. Dung, "Generating Finite-State Transducers for Semi-Structured Data Extraction from the Web", *Information Systems*, Vol. 23(8), 1998, pp. 521–538.

[23] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features", in *Proceedings of the 10th European Conference on Machine Learning*, Chemnitz, Germany, 1998, pp. 137–142.

[24] D. Koller and M. Sahami, "Hierarchically classifying documents using very few words", in *Proceedings of the Fourteenth International Conference on Machine Learning*, 1997, pp. 170–178.

[25] N. Kushmerick, "Wrapper Induction: Efficiency and Expressiveness", *Artificial Intelligence*, Vol. 118 (1-2), 2000, pp. 15–68.

[26] A. H. F. Laender, B. A. Ribeiro-Neto, A. S. da Silva, and Juliana S. Teixeira, "A Brief Survey of Web Data Extraction Tools", *SIGMOD Rec.*, Vol. 31 (2), 2002, pp. 84–93.

[27] W. Lam and C.Y. Ho. "Using a Generalized Instance set for Automatic Text Categorization", in *Proceedings of the 21th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1998, pp. 81–89.

[28] L. S. Larkey and W. B. Croft. "Combining classifiers in text categorization", in *Proceedings of the 19th Annual international ACM SIGIR Conference on Research and Development in information Retrieval*

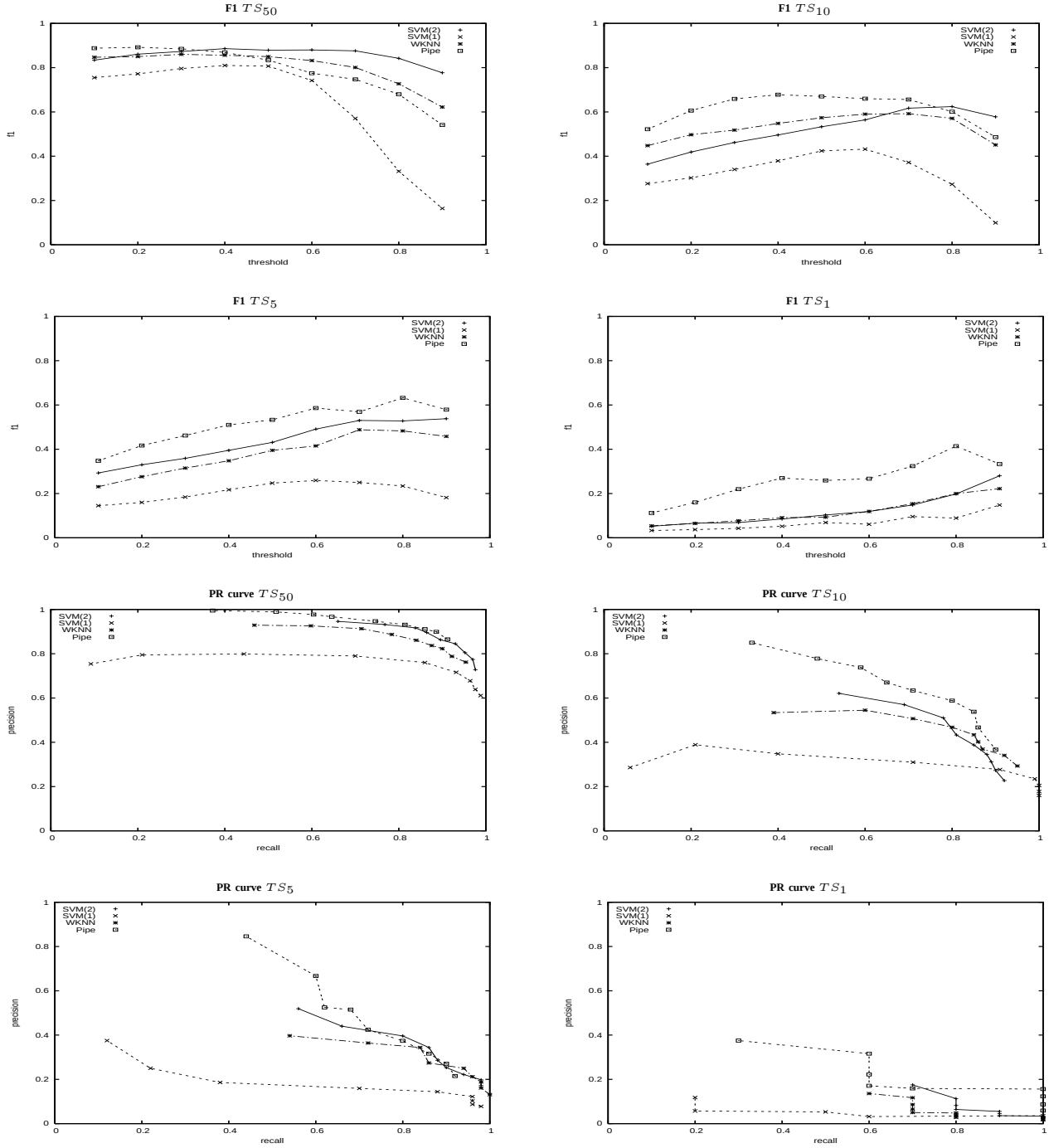

Fig. 5. Experimental results performed on a specific pipeline, i.e. [ECAT, E21, E211], reported together those obtained by running other classifiers on the flat –i.e. non hierarchical– model.

(Zurich, Switzerland, August 18 - 22, 1996), ACM Press, New York, NY, 1996, pp. 289–297.

[29] D.D. Lewis and M. Ringuette, "A Comparison of Two Learning Algorithms for Text Categorization", in *Proceedings of the 3rd Symposium on Document Analysis and Information Retrieval*, Las Vegas, NV, USA, 1994, pp. 81–93.

[30] D.D. Lewis, R. E. Schapire, J. P. Callan, and R. Papka, "Training Algorithms for Linear Text Classifiers", in *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, (Zurich, Switzerland, August 18 - 22, 1996). SIGIR '96. ACM Press, New York, NY, 1996, pp. 298–306.

[31] D.D. Lewis, Y. Yand, T. Rose, F. Li, "Rcv1: A New Benchmark Collection for Text Categorization Research", in *Journal of Machine Learning Research*, Vol. 5(Dec.2004), 2004, pp. 361–397.

[32] A. McCallum and K. Nigam, "A Comparison of Event Models for Naive Bayes Text Classification", in *Proceedings of AAAI-98 Workshop on Learning for Text Categorization*, AAAI Press, 1998, pp. 359–367.

[33] B. Masand, G. Lino, and D. Waltz. "Classifying News Stories Using Memory Based Reasoning". In *Development in Information Retrieval*, N. J. Belkin, P. Ingwersen, and A. M. Pejtersen, editors, Kobenhavn, DK: ACM Press, New York, US, 1992, pp. 59–65.

[34] I. Moulinier, G. Raskinis, and J.G. Ganascia, "Text categorization: a Symbolic Approach", in *Proceedings of 5th Annual Symposium on Document Analysis and Information Retrieval*, Las Vegas, US, 1996, pp. 87–99.

[35] I. Moulinier, "Is Learning Bias an Issue on the Text Categorization Problem?", in Technical report, LAFORIA-LIP6, Universite Paris VI, 1997.

[36] H.T. Ng, W.B. Goh and K.L. Low, "Feature Selection, Perceptron Learning, and a Usability Case Study for text Categorization", in *Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval*, July 27-31, Philadelphia, 1997, pp. 67–73, .

[37] M. Porter, "An Algorithm for Suffix Stripping", *Program*, Vol. 14(3), 1980, pp. 130-137.

[38] A. Sahuguet and F. Azavant, "Building Intelligent Web Applications Using Lightweight Wrappers", *Data Knowledge Engineering*, Vol. 36(3), 2001, pp. 283–316.

[39] R.E. Schapire, Y. Singer and A. Singhal, "Boosting and Rocchio Applied to Text Filtering", in *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1998, pp. 215–223.

[40] Robert E. Schapire, "Theoretical views of boosting", In *Computational Learning Theory: Fourth European Conference, EuroCOLT'99*, 1999, pp. 1–10.

[41] F. Sebastiani, A. Sperduti, and N. Valdambrini, "An Improved Boosting Algorithm and its Application to Text Categorization", in *Proceedings of the Ninth international Conference on information and Knowledge Management* (McLean, Virginia, United States, November 06 - 11, 2000), ACM Press, New York, NY, 2000, pp. 78–85.

[42] F. Sebastiani, "Machine learning in automated text categorization", *ACM Computing Surveys*, 34(1) (Mar. 2002), 2002, pp. 1–47.

[43] S. Soderland, "Learning Information Extraction Rules for Semi-Structured and Free Text", *Machine Learning*, Vol. 34(1-3), 1999, pp. 233–272

[44] K. Tzeras and S. Hartmann, "Automatic Indexing Based on Bayesian Inference Networks", in *Proceedings of SIGIR'93, 16th ACM International Conference on Research and Development in Information Retrieval*, 1993, pp. 22–34.

[45] C. van Rijsbergen, *Information Retrieval*, Butterworths, London, 1979.

[46] A.S. Weigend, E.D. Wiener, and J.O. Pedersen, "Exploiting Hierarchy in Text Categorization", *Information Retrieval*, 1(3), 1999, pp. 193–216.

[47] A. S. W. Erik Wiener, J. O. Pedersen, "A Neural Network Approach to Topic Spotting", in *Proceedings of 4th Annual Symposium on Document Analysis and Information Retrieval*, 1995, pp. 317–332.

[48] Y. Yang and C. Chute, "An Example-Based Mapping Method for Text Categorization and Retrieval". *ACM Transactions on Information Systems*, Vol. 12(3), 1994, pp. 252–277.

[49] Y. Yang, "Expert Network: Effective and Efficient Learning from Human Decisions in Text Categorization and Retrieval", in *Proceedings of 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Dublin, Ireland, July 03 - 06, 1994). W. B. Croft and C. J. van Rijsbergen, Eds. Annual ACM Conference on Research and Development in Information Retrieval. Springer-Verlag New York, New York, NY, 1994, pp. 13–22.

[50] Y. Yang, "An Evaluation of Statistical Approaches to Text Categorization", in *Information Retrieval*, Vol. 1(1-2), 1999, pp. 69–90.

[51] Y. Yang and X. Liu, "A re-examination of text categorization methods", in *Proceedings of the 22nd Annual international ACM SIGIR Conference on Research and Development in information Retrieval*, (Berkeley, California, United States, August 15 - 19, 1999). SIGIR '99. ACM Press, New York, NY, 1999, pp. 42–49.

[52] Y. Yang, T. Ault and T. Pierce, "Combining multiple learning strategies for effective cross-validation", in *Proceedings of the 17th International Conference on Machine Learning*, 2000, pp. 1167-1182.

[53] Z. Zheng, "Naive Bayesian Classifier Committees", in *Proceedings of the 10th European Conference on Machine Learning* (April 21 - 23, 1998). C. Nedellec and C. Rouveirol, Eds. Lecture Notes In Computer Science, Vol. 1398. Springer-Verlag, London, 1998, pp. 196–207.

Giuliano Armano obtained his Ph.D. in Electronic Engineering from the University of Genoa, Italy, in 1990. He is currently associate professor of computer engineering at the Dept. of Electrical and Electronic Engineering (DIEE), University of Cagliari, leading also the IASC ("Intelligent Agents and Soft-Computing") group. His educational background ranges over expert systems and machine learning, whereas his current research activity is focused on (i) proactive and adaptive behavior of intelligent agents as well as on (ii) hybrid genetic-neural architectures and systems. The above research topics are mainly experimented in the field of bioinformatics, information retrieval, and text categorization.

Francesco Mascia is a Ph.D. Student in Computer Engineering at the University of Cagliari, Italy. He received the MS degree in Electronic Engineering from the University of Cagliari, Italy, in 2006. His research interests are currently in the fields of machine learning, bioinformatics, and text categorization.

Eloisa Vargiu obtained her Ph.D. in Electronic and Computer Engineering from the University of Cagliari, Italy, in 2003. Since 2000 she collaborates with the Intelligent Agents and Soft-Computing (IASC) Group at the Dept. of Electrical and Electronic Engineering (DIEE), University of Cagliari. Her educational background is mainly focused on intelligent agents, in particular on their proactive and adaptive behavior. The above research topics are mainly experimented in information retrieval, text categorization, and bioinformatics.