A Model for Term Selection in Text Categorization Problems

Laura Maria Cannas, Nicoletta Dessì and Stefania Dessì
Dipartimento di Matematica e Informatica
Università degli Studi di Cagliari
Cagliari, Italy
{lauramcannas, dessi, dessistefania}@unica.it

***Abstract*—In the last ten years, automatic Text Categorization (TC) has been gaining an increasing interest from the research community, due to the need to organize a massive number of digital documents. Following a machine learning paradigm, this paper presents a model which regards TC as a classification task supported by a wrapper approach and combines the utilization of a Genetic Algorithm (GA) with a filter. First, a filter is used to weigh the relevance of terms in documents. Then, the top-ranked terms are grouped in several nested sets of relatively small size. These sets are explored by a GA which extracts the subset of terms that best categorize documents. Experimental results on the Reuters-21578 dataset state the effectiveness of the proposed model and its competitiveness with the learning approaches proposed in the TC literature.**

***Keywords*-text categorization, term selection, hybrid model, genetic algorithm**

I. INTRODUCTION

Text categorization (TC) is the study of assigning natural language documents to one or more predefined category labels. Because of the need to automatically organize the increasing number of digital documents in flexible ways, TC is receiving a crescent interest from researchers and developers.

The dominant approach to this problem considers the employment of a general inductive process that automatically builds a classifier by learning, from a set of pre-classified documents, the characteristics of the categories [1]. Many information retrieval, statistical classification and machine learning techniques have been applied to TC domains. Examples are Rocchio's algorithm [1], regression models [2], K-nearest neighbor [2], Naïve Bayes [3], SVM [3][4][5], Decision trees (e.g. C4.5 decision tree algorithm [5]), and neural networks [6] etc.

However, most algorithms may not be completely suitable when the problem of high dimensionality occurs [2][4], as even a moderately sized text collection often has tens of thousands of terms which make the classification cost prohibitive for many learning algorithms that do not scale well to large problem sizes. In addition, it is known that most terms are irrelevant for the classification task and some of them even introduce noise that may decrease the overall performance [7].

Applying dimensionality reduction techniques (i.e. feature selection or feature extraction) is beneficial for the increasing scalability, reliability, efficiency and accuracy of text classification algorithms [8]. In this paper, we consider term selection, i.e. the feature selection process that reduces the dimensionality of the feature space by only retaining the most informative or discriminative terms.

Generally, feature selection algorithms can be broadly divided in two categories: filters and wrappers. Filter approaches evaluate the relevance of each single term according to a particular feature scoring metric and retain the best t terms. Although simple and fast, filters lack robustness against correlations between terms and it is not clear how to determine the optimal values of t, namely the threshold value. Conversely, wrappers compare different term subsets and evaluate them using the classification algorithm that will be employed to build the final classifier. To have an exhaustive search, in practice, greedy procedures or meta-heuristics are usually employed to guide a combinatorial search through the space of candidate term subsets looking for a good trade-off between performance and computational cost. Even if wrapper methods have been shown to generally perform better than filters [4], their time-consuming behaviour has made the use of filter approaches in the TC area prominent.

In this paper we present a hybrid model for term selection which combines and takes advantage of both filter and wrapper approaches in order to overcome their limitations.

In detail, the model uses a filter to rank the list of terms present in documents. Then, terms with the highest score values are selected, in an incremental way, resulting in a set of nested term subsets. The preliminary use of the filter ensures that useful terms are unlikely to be screened out. Differently from most filter-based approaches, the ranked list is not cut off according to a single (somewhat arbitrary) threshold value. To limit classification problems due to the correlation among terms, our approach considers refining the selection process by employing a wrapper that uses a Genetic Algorithm (GA) as a search strategy. Unlike traditional wrappers that select the features linearly, a GA performs a random terms combination and shows its potentiality in exploring features set of high dimensionality.

The above described model is named the Genetic Wrapper Model (GWM).

To evaluate the proposed approach we choose the standard test sets Reuters-21578 [9]. Experimental results compare well with some of the top-performing learning algorithms for TC and confirm the effectiveness of our model.

The rest of the paper is organized as follows. Section 2 details the proposed model. The experimental analysis and the related results are presented in Section 3. Finally, conclusions are outlined in Section 4.

II. THE PROPOSED MODEL

Formally, a problem of TC can be defined as follows. Let $D = \{d_1, d_2, \ldots, d_N\}$ be a collection of N documents and $W = \{w_1, w_2, \ldots, w_M\}$ be a set of M distinct terms contained in D. Let $C = \{ c_1, c_2, \ldots, c_{|C|} \}$ be a set of predefined categories or classes. A TC process assigns a boolean value to each pair $<d_j, c_i>$ and indicates if the document d_j belongs to the category c_i.

In a multi-label TC problem each document can be assigned to any number of categories from the set C. Under the assumption that categories are stochastically independent of each other, a multi-label TC can be transformed into |C| independent (disjoint) binary TC problems, where each document is classified in one of the two disjoint categories: c and its complement $\bar{c}$. Therefore, to solve a multi-label TC problem, binary classifiers are built for each category in C and their results are then combined into a single decision.

Likewise, in this work we address a multi-label TC problem by resolving |C| binary problems. Our GWM first selects the most representative terms for a given category c_i and then performs a binary classification process on this selection.

Figure 1 shows the basic steps of GWM. The model input, i.e. the training set, is a matrix where each row represents a document d_j and columns are the related terms $\{w_1, w_2, \ldots, w_M\}$. Each document is assigned to either the category c_i or its complement $\bar{c}_i$.

First, a filter method assesses the scores of individual terms according to their power in discriminating c_i. This results in an ordered list where terms appear in descending order of relevance. The aim is to guide the term research at the initial stage and ensure that useful terms are unlikely to be discarded.

With a fixed threshold value R, different term subsets of increasing size, namely Building Blocks (BBs), are constructed by progressively adding to the first R terms of the ordered list, additional terms are less and less correlated with the category. It results in a sequence of Q nested BBs:

$$BB_1 \subset BB_2 \subset BB_3 \subset \ldots \subset BB_Q$$

where BB_1 includes the first R top-ranked terms, BB_2 includes the first 2*R top-ranked term, etc.

Then, such BBs are refined by a wrapper that uses a GA as a search strategy, with the intent of removing redundant terms and obtaining more accurate and small-sized subsets of terms for categorization. Specifically, for each BB, the GA randomly initializes a population of individuals, each individual being codified by a binary vector whose dimension equals the size of the BB. In the binary vector, the value 1 means that the respective term is selected, otherwise the value is 0. A fitness function evaluates the individuals by means of a classifier and selects the individuals that maximize the classification accuracy. Then, the current population undergoes genetic operations (i.e. selection, mutation, and crossover) and a new population is generated and evaluated. This evolution process is repeated within a pre-defined number of generations and it outputs the best individual, i.e. the subset of terms that best categorizes the BB.

Using a test set, solutions from each BB are evaluated and compared using popular metrics. The solution that shows the highest value of the considered metrics is selected as the best one and is returned by the GWM. This solution is the subset of terms that best categorizes the given category c_i.

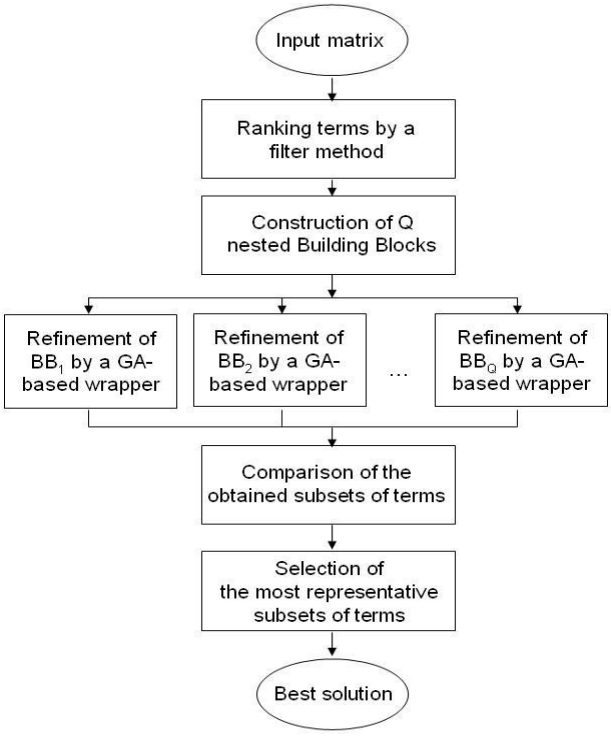

Figure 1. Steps of the proposed Genetic Wrapper Model

III. MODEL EVALUATION

A. Experimental Setup

The model is generic and its evaluation can be supported by a variety of popular filter techniques, in the same way as different classifiers can be employed in conjunction with the GA within the wrapper.

For ranking, we experimented with the filters χ^2 (CHI) and Information Gain (IG). Hence, we implemented two versions of GWM that differ in the choice of the filter technique, namely GWM(CHI) and GWM(IG).

To build the nested BBs, we set R=10 and Q=10 in what we considered the first 100 top-ranked terms. We also evaluated two additional BBs with the sizes 150 and 200.

The wrapper is based on the GA search mechanism as proposed by Goldberg [10]. Leveraging on previous studies about tuning GA parameters [11], we set the following values: population size = 30, crossover probability = 1, mutation probability = 0.02, number of generations = 50. Since the GA performs a stochastic search, we considered the results from 3 trials. The fitness function used the Naïve Bayes Multinomial classifier [12] for accuracy estimation.

The evaluation and comparison of the solutions obtained from each BB were performed by the following popular metrics [1]: F-measure, which expresses the harmonic mean between precision and recall, Break Even Point (BEP), which expresses the mathematical mean between precision and recall, and µ-BEP, which permits a global evaluation of BEP values across categories.

The overall analysis was implemented using the Weka data mining environment [13].

We tested the proposed model on the Reuters-21578 test collection that consists of 12,902 documents clustered in 135 categories. We used the Mod-Apté split, where 9,603 documents are used as a training set and the remaining 3,299 documents form the test set. We used the dataset as pre-processed in [14], which considers the 10 categories with the highest number of positive training examples. In the following we will refer to this subset as R10. Table I shows the number of terms for each category in R10.

TABLE I. DIMENSIONALITY OF CATEGORIES IN R10

Category	Number of terms
acq	7,495
corn	8,302
crude	14,466
earn	9,500
grain	12,473
interest	10,458
money-fx	7,757
ship	9,930
trade	7,600
wheat	8,626

B. Results and Discussion

In this section we describe the experimental results obtained by GWM(IG) and GWM(CHI).

For each BB, we compared results on 3 trials and chose the solution with the highest F-measure as the best one. As Table II shows, the best solution, in terms of both F-measure and number of selected terms, does not significantly differ from the relative average values. Table II only details results obtained by GWM(IG) for the category grain, similar trends have been noticed for all the categories irrespective of the implementation of GWM.

For this reason, in the following we will consider and report only the best F-measure values.

Figure 2 shows the best value of F-measure obtained by the proposed model within each BB. The GWM(IG) version results in very high values of F-measure compared to those obtained by GWM(CHI).

For each BB, Figure 3 shows the size of the best solution expressed by the rate between the terms selected by the model and the respective size of the initial BB.

The above results demonstrate that GWM(IG) is more specific than GWM(CHI) as it allows to select a lower number of terms.

TABLE II. AVERAGE AND BEST VALUES OBTAINED WITH GWM(IG) ON CATEGORY GRAIN

	Average Values		Best Values	
BB size	**F-measure**	**Selected Terms**	**F-measure**	**Selected Terms**
10	53.16	9	53.16	9
20	65.48	18	65.48	18
30	92.28	12	92.78	13
40	91.59	15	92.45	14
50	91.16	16	91.56	17
60	90.77	19	92.26	17
70	90.30	24	91.66	21
80	89.03	24	90.36	24
90	89.61	30	91.61	29
100	90.09	27	92.26	19
150	89.70	46	92.26	36
200	89.16	63	90.37	58

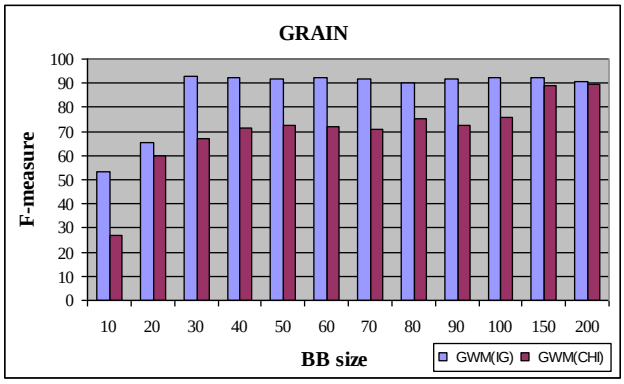

Figure 2. Best F-measure values obtained within each BB (category grain)

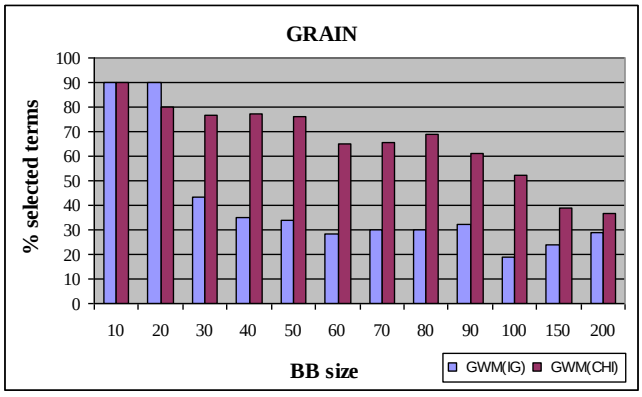

Figure 3. Percentage of selected terms from each BB (category grain)

TABLE III. BEST F-MEASURE VALUE AND RELATED BEP VALUE OBTAINED FOR EACH CATEGORY IN R10

	GWM(IG)					GWM(CHI)				
Category	**BB size**	**Selected Terms**	**F-measure**	**BEP**	**Time (sec)**	**BB size**	**Selected Terms**	**F-measure**	**BEP**	**Time (sec)**
acq	200	105	90.36	**90.40**	115	200	107	88.46	88.55	117
corn	150	30	93.09	**93.20**	116	200	123	56.52	62.85	108
crude	50	33	86.52	**86.85**	64	200	111	79.91	80.75	95
earn	150	73	96.90	96.90	124	200	97	97.05	**97.05**	129
grain	30	13	92.79	**92.85**	60	200	73	89.82	90.05	115
interest	90	34	60.68	**60.70**	96	200	110	58.29	58.35	113
money-fx	150	69	66.51	**66.95**	125	200	111	63.21	63.70	148
ship	90	47	84.09	**84.10**	106	200	122	70.74	74.05	95
trade	60	30	67.29	**67.70**	77	200	101	60.48	63.00	110
wheat	40	5	90.81	**91.20**	75	150	98	59.29	65.80	92

To detail model results, we report in the following the best solution obtained by GWM(IG) over the category grain, i.e. the subset of terms that best categorizes this category:

{ *wheat, grain, tonnes, corn, maize, barley, rice, cts, program, company, shr, commodity, bushel* }.

Table III compares the performance of the two GWM implementations obtained for the categories in R10. It shows that GWM(IG) is more effective than GWM(CHI) in all the categories, with the exception of the category earn. By using a different scale for F-measure, Figure 4 shows this small abnormal behaviour.

Furthermore, Table III illustrates the performance of the proposed model in terms of BEP and computational time (using a 3.6 GHz AMD Phenom 4 GB RAM).

From Table III, Figure 5 shows the comparison between the F-measure values obtained by using GWM(IG) and GWM(CHI).

In [14], two similar models, namely Olex-GA and Olex Greedy [15], are proposed and evaluated on R10. Additionally, the best results in [14] are compared with the best values obtained by the following classifiers: Naïve Bayes, C4.5, Ripper, and SVM (both, polynomial and radial basis function - rbf).

Table IV shows how results, as reported in [14], compare well with the best results obtained from our model in terms of BEP and μ-BEP values. Indeed, with a μ-BEP of 89.06, our model outperforms Naïve Bayes (82.52), Olex Greedy (84.80), C4.5 (85.82), Olex-GA (86.40), Ripper (86.71), and SVM rbf (88.80) and is competitive with SVM poly (89.91). Although the comparison is based on the best results, for the sake of completeness Table IV reports in brackets the corresponding average BEP values obtained from our model.

IV. CONCLUSIONS

In this paper we presented a model supporting TC problems. Specifically, our model selects the most representative terms for a given category and then performs a classification process on this selection. An extensive validation has been carried out on the standard data collection Reuters-21578 and experimental results confirm the effectiveness of our model. In fact, it compares well with several learning algorithms used in the TC domain.

From a machine learning point of view, TC is a challenging research area as datasets consist of hundreds of thousands of documents and are characterized by tens of thousands of terms. This means that TC is a good benchmark for checking whether our model can scale up to substantial sizes. Moreover, the proposed model does not fall squarely under the classes of algorithms that are usually adopted to solving TC problems. Although many approaches have been proposed in TC literature, GA-based learning approaches have remained isolated attempts.

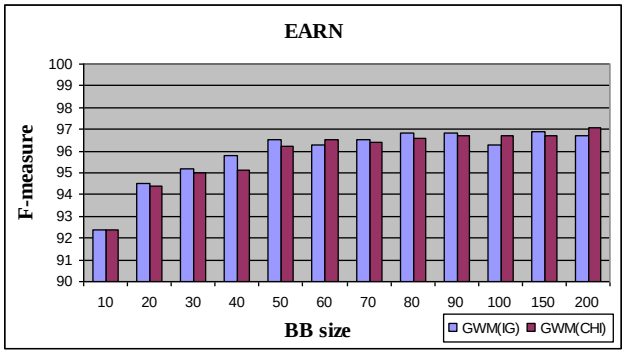

Figure 4. Best F-measure values obtained within each BB (category earn)

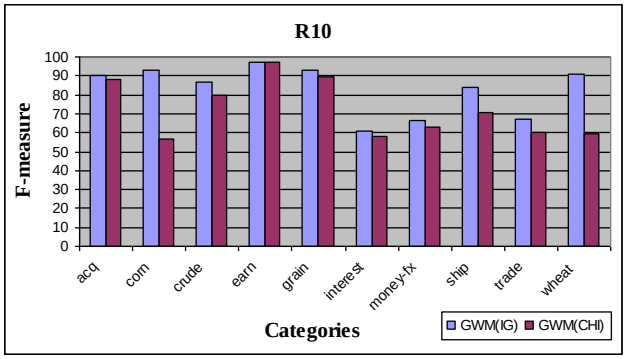

Figure 5. Best F-measure values (in R10)

TABLE IV. COMPARISON USING BEP AND μ–BEP VALUES

Category	Naïve Bayes	C4.5	Ripper	SVM		Olex		GWM
				Poly	rbf	Greedy	GA	
acq	90.29	85.59	86.63	90.37	90.83	84.32	87.49	90.40 (89.93)
corn	59.41	86.73	91.79	87.16	84.74	89.38	91.07	93.20 (87.93)
crude	78.84	82.43	81.07	87.82	86.17	80.84	77.18	86.85 (83.58)
earn	96.61	95.77	95.31	97.32	96.57	93.13	95.34	97.05 (96.70)
grain	77.82	89.69	89.93	92.47	88.94	91.28	91.75	92.85 (92.35)
interest	61.71	52.93	63.15	68.16	58.71	55.96	64.59	60.70 (59.03)
money-fx	56.67	63.08	62.94	72.89	68.22	68.01	66.66	66.95 (63.82)
ship	68.68	71.72	75.91	82.66	80.40	78.49	74.81	84.10 (82.30)
trade	57.90	70.04	75.82	77.77	74.14	64.28	61.81	67.70 (64.33)
wheat	71.77	91.46	90.66	86.13	89.25	91.46	89.86	91.20 (90.78)
μ-BEP	**82.52**	**85.82**	**86.71**	**89.91**	**88.80**	**84.80**	**86.40**	**89.06 (88.03)**

As such, our proposal seems to offer several research perspectives. First, results show that the hybrid approach used for term selection combines effectiveness and efficiency as the initial use of a filter permits to reduce the computational cost of the GA- based wrapper.

Second, we note that the choice of the specific filter is significant, as it notably influences the model performance. Our results confirm what has been already observed in literature [4]: sometimes CHI presents erratic behaviour in the TC domain. In contrast, in this study IG turned out to be incisive in conjunction with the evolutionary wrapper.

In our future work, the proposed model will be evaluated further by considering different ranking methods for weighing terms as well as different values for building the nested BBs.

ACKNOWLEDGMENT

This research was supported by RAS, Regione Autonoma della Sardegna (Legge regionale 7 agosto 2007, n. 7 "Promozione della ricerca scientifica e dell'innovazione tecnologica in Sardegna") in the project "DENIS: Dataspaces Enhancing the Next Internet in Sardinia".

Stefania Dessì gratefully acknowledges Sardinia Regional Government for the financial support of her PhD scholarship (P.O.R. Sardegna F.S.E. Operational Programme of the Autonomous Region of Sardinia, European Social Fund 2007-2013 - Axis IV Human Resources, Objective l.3, Line of Activity l.3.1.)".

REFERENCES

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

[2] Yang, Y. and Pedersen, J. O. A comparative study on feature selection in text categorization. Proceedings of ICML-97, 14th International Conference on Machine Learning, ed. D.H. Fisher, Morgan Kaufmann Publishers, San Francisco, US: Nashville, US, pp. 412–420 (1997)

[3] Wang, G. and Lochovsky, F. H. Feature selection with conditional mutual information maximin in text categorization. Proceedings of the thirteenth ACM international conference on Information and knowledge management, pp. 342-349 (2004)

[4] Forman, G. An Extensive Empirical Study of Feature Selection Metrics for Text Classification. Journal of Machine Learning Research, 3, 1289-1305 (2003)

[5] Joachims, T. Text categorization with support vector machines: learning with many relevant features. Proc. of ECML-98, 10th European Conference on Machine Learning(Chemnitz, Germany), 137-142 (1998)

[6] Yang, Y. and Liu, X. A re-examination of text categorization methods. In Proceedings of SIGIR-99, 22nd ACM International Conference on Research and Development in Information Retrieval (Berkeley, CA), 42-49 (1999)

[7] Salton, G. and McGill, M. J. An Introduction to Modern Information Retrieval. McGrawHill (1983)

[8] Lewis, D. D. Feature Selection and Feature Extraction for Text Categorization. Proc. Speech and Natural Language Workshop, pp. 212-217 (1992)

[9] Lewis, D.D. Reuters-21578 text categorization test collection. Distribution 1.0 (1997)

[10] Goldberg, D.E. Genetic algorithms in search, optimization and machine learning. Addison-Wesley (1989)

[11] Cannas, L.M., Dessì, N., Pes B. A filter-based evolutionary approach for selecting features in high-dimensional micro-array data. IIP 2010, IFIP AICT 340, 297--307 (2010)

[12] Mccallum, A., Nigam, K. A Comparison of Event Models for Naive Bayes Text Classification. AAAI-98 Workshop on 'Learning for Text Categorization' (1998)

[13] Hall, M., et al. The WEKA data mining software: an update. SIGKDD Explorations, vol. 11, no. 1 (2009)

[14] Pietramala, A., Policicchio, V.L., Rullo, P., Sidhu, I. A Genetic Algorithm for Text Classification Rule Induction. ECML/PKDD (2): 188-203 (2008)

[15] Rullo, P., Policicchio, V.L., Cumbo, C., Iiritano, S. Olex: effective rule learning for text categorization. IEEE TKDE , vol. 21, no. 8, pp. 1118-1132 (2009)