A Multistrategy Approach for Digital Text Categorization from Imbalanced Documents

M. Dolores del Castillo
Instituto de Automática Industrial (CSIC)
Ctra. Campo Real Km. 0.200, 28500 Arganda del Rey
Madrid. SPAIN
lola@iai.csic.es

José Ignacio Serrano
Instituto de Automática Industrial (CSIC)
Ctra. Campo Real Km. 0.200, 28500 Arganda del Rey
Madrid. SPAIN
nachosm@iai.csic.es

ABSTRACT

The goal of the research described here is to develop a multistrategy classifier system that can be used for document categorization. The system automatically discovers classification patterns by applying several empirical learning methods to different representations for preclassified documents belonging to an imbalanced sample. The learners work in a parallel manner, where each learner carries out its own feature selection based on evolutionary techniques and then obtains a classification model. In classifying documents, the system combines the predictions of the learners by applying evolutionary techniques as well. The system relies on a modular, flexible architecture that makes no assumptions about the design of learners or the number of learners available and guarantees the independence of the thematic domain.

Keywords

Feature selection, multistrategy learning, genetic algorithms.

1. INTRODUCTION

Text categorization can be applied in any context requiring document organization or selective and adaptive document dispatching. Assigning thematic categories to documents is essential to the efficient management and retrieval of information and knowledge [25]. This paper focuses on the task of classifying incoming documents in several non-disjoint categories.

Although the growth of electronically stored text has led to the development of machine learning methods prepared to exploit ungrammatical text, most of these methods are based on a single strategy and work well in concrete domains [9], [16]. The richness and redundancy of the information present in many digital documents make a multistrategy learning approach especially suitable [8]. However, most current multistrategy systems for text classification [7], [10] combine statistical and symbolic algorithms in a predefined manner by using a common feature extraction stage and thus a shared feature set. These systems solve the problem by different empirical methods and usually take the most confidential method.

Certain learning algorithms are more suitable for some thematic categories than for others [10], showing different classification results due to the different types of information present in each domain. The performance of an algorithm depends on the features or attributes chosen to represent the information [12], [15], [26]. Choosing the right feature set is critical to the successful induction of classification models [20]. Conventional approaches use a general method based on statistical measurements and stemming procedures for creating the feature set or vocabulary of the problem, which is independent of the learning algorithm and the thematic domain [18]. In [2] several experiments were carried out to monitor the actual interaction between feature selection and the performance of some linear classifiers.

The algorithm used and the features selected are always the key points at design time, and many experiments are needed to select the final algorithm and the best suited feature set. Moreover, once the algorithm and features are set, the achieved solution may prove unsatisfactory due to possible losses of relevant information when mapping from documents to the feature set.

The main goal of the HYCLA (HYbrid CLAssifier) system presented here is to maximize classification performance by considering all the types of information contained in documents regardless of their thematic domain. With this aim, the classification system relies on a hybrid architecture that tackles two main issues: optimization of document representation and integration of the results of several classifiers. The term *hybrid* has a double meaning here. On one hand, it symbolizes the multistrategy nature of the empirical learning approach to text categorization. On the other, it refers to the genetic search carried out to find the vocabulary of the problem and integrate the individual predictions of the learners.

HYCLA learns classification models from imbalanced document samples. The documents can be imbalanced for two reasons: 1) some thematic categories have many preclassified documents, while others do not; and 2) there are thematic categories that only contain one or two types of information.

Feature selection methods based on a particular statistical measurement favor some thematic categories over others depending on the characteristics of the ranking statistical technique and the thematic categories. The genetic feature selection proposed in this paper treats all categories the same because it considers several statistical measurements, thus obviating the kind of imbalance that stems from a different distribution in the number of documents per category.

HYCLA distinguishes several types of text information present in documents and builds a classification model for each type of information. When it classifies a document, the final document category is obtained by the genetic combination of the decisions made by all the models. Since there are many web domains containing thematic categories that lack some part of information, the documents belonging to these categories could yield worse results in classification. The goal of the genetic combination is to smooth out this other kind of imbalance by optimizing the

contribution each classification model makes towards assigning the final category to a document.

The HYCLA system has been validated using two types of digital or electronically stored text: scientific/technical papers and hypertext documents belonging to several categories.

The following section surveys the architecture capabilities in detail. Section 3 discusses the empirical evaluation of this approach, and final sections present the conclusions and point the way to future work on this subject.

2. SYSTEM ARCHITECTURE

HYCLA operates in two stages, learning and integration. In the learning stage, learners apply an evolutionary technique to obtain their own feature set, and then they are trained to obtain their classification model. In the integration stage, individual learned models are evaluated on a test set, and the predictions made are combined in order to achieve the best classification of test documents. The subsections below describe the modules and procedures of this system.

The underlying architecture of HYCLA can be instantiated to approach a different text mining task by upgrading its modules.

2.1. Preprocessing Step

This step is common to all the learners. The system receives a sample of documents of different thematic categories that is divided into two sets, the training set which contains two-thirds of the documents and the test set which contains one-third of the documents. The task here is to scan the text of the sample and produce the list of the words or vocabulary contained in the documents.

Figure 1 shows the analogies found between the parts of scientific and hypertext documents. These documents usually present redundant information in all four of their text parts [1].

Based on this idea, when the system receives a training sample of scientific/hypertext documents whose first line is the title/*url* (*u*niform *r*esource *l*ocator) of the document, four vocabularies are generated from every document: one containing the title/*url* words, a second containing all the words from the from abstract/meta-text, a third with the contents/plain words, and a fourth containing the words from the references/hyperlinks. Every vocabulary is smaller in size than the vocabulary obtained from the original document. Whenever all the documents lack some portion of the text, the corresponding vocabulary is empty.

The preprocessing step begins by removing those words found in the vocabularies that belong to a stop list consisting of words without semantic content [12], [15] and applying stemming procedures [22]. After that, the frequency of occurrence of every valid word is calculated, and this value is increased depending on the word format (for example, the frequency is ten times higher for a word found in the title, nine times higher for a word found in the subtitle, and so on). Words recurring below a frequency threshold are not reliable indicators and are removed.

Due to the high dimensionality of the preprocessed vocabularies the first task of HYCLA is to reduce the feature space size with the lowest loss of classification performance. For each preprocessed vocabulary, once the number of documents from every category containing the terms of the vocabulary is known, several information statistical measurements are calculated: 1) information gain: how many information bits are needed to predict a category depending on the presence/absence of a word in a document [26]; 2) mutual information: words occurring only in a document belonging to a certain category are the most relevant for this category [26]; 3) document frequency: words occurring more frequently are the most valuable; 4) chi square: how independent a word and a category are [24]; 5) crossover entropy: similar to information gain, but considering the presence of a word in a document [24]; and 6) odds-ratio: the relevance of a

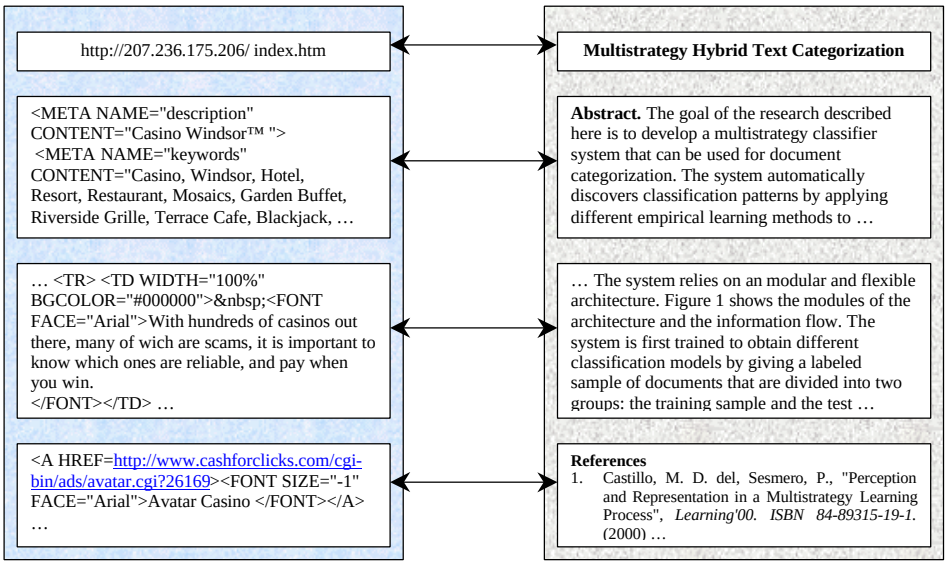

Figure 1. Structural analogy between scientific papers and HTML documents.

word for a category depends on the occurrence and non-occurrence of the word in the category [20]. The values of some of these six measurements depend heavily on the distribution of documents into thematic categories. An imbalanced document sample contributes to strengthen the value of only some of these statistical measurements.

The words of all of the vocabularies are sorted by the six measurements, and only the k_v highest ranked words of each vocabulary are retained. In [25], [20] a detailed analysis of the optimal value of k_v is discussed. The authors show that the relation between some measures of classification performance and k_v depends heavily on the particular statistical measurement chosen and whether the classifier is binary (disjoint categories) or multicategory (non-disjoint categories). Since the genetic feature selection carried out by HYCLA considers several statistical measurements, the way to avoid these dependences is to adopt a value of k_v of approximately 30% of the size of the preprocessed vocabulary. All the statistical measurements achieve their maximum performance classification with this value of k_v.

The k_v words of each vocabulary ranked by each measurement form a view. If several views of a vocabulary are identical, then only one of them is considered. The set of views of a vocabulary will be the initial feature subsets of a learner.

Although some information is lost in any one feature subset, the multiple views of every initial vocabulary will make for a better overall performance. In scientific/hypertext documents, there are four possible vocabularies and six possible views associated with each vocabulary.

2.2. Learners: Structure and Dynamics

Since documents contain different kinds of information, the multistrategy approach suggests that each learner solves a part of the problem with a different incoming information from the same sample. Each learner can learn to classify documents with regard to the feature subsets obtained from the preprocessing step.

The filtering feature selection by ranking techniques is independent of the learning method that will use the selected features. The performance of a learner using filtered features is very sensitive to the score criterion of the ranking technique. In order to avoid this situation, HYCLA adopts what has been called the wrapper approach [26], [13], in which final feature selection depends on the inductive algorithm used.

When a learner receives a feature set, it carries out the following tasks:

1. Empirical learning

- *Feature selection.* Every learner applies a genetic algorithm to achieve an optimal feature set in a large, criterion independent search space.
- *Classification model.* The learner works on the training documents, represented according to the feature set learned, to induce the classification model.

2. Testing. The learner applies the inferred model to a test set and calculates several measures of classification performance.

2.2.1. Genetic Feature Selection

Genetic algorithms are search algorithms based on the natural evolution process. They have been successfully applied to optimization and machine learning problems [3], [11]. Starting from an initial population of individuals or chromosomes representing tentative solutions to a problem, a new generation is created by combining or modifying the best individuals of the previous generation. The process ends when the best solution is achieved or after a fixed number of generations.

The application of genetic algorithms to text feature selection involves establishing the representation of chromosomes, defining crossover and mutation operators fitted to chromosome representation and document domain, and defining the fitness function used to determine the best chromosomes of a population.

The goal of genetic feature selection is to solve the kind of imbalance derived from a different distribution in number of documents per category

2.2.1.1. Chromosome Representation

Each view computed from an original vocabulary in the preprocessing step is a chromosome. Chromosome length is fixed at k_v. Each gene is a word of the vocabulary. Population size matches the number of different views of a vocabulary. For example, if the input vocabulary is {*bye, see_you, hello, good_morning, good_afternoon*}, then {*see_you, bye, good_afternoon*} and {*see_you, good_afternoon, hello*} are two chromosomes that could be obtained by applying chi-square and crossover entropy techniques, respectively, with $k_v = 3$.

2.2.1.2. Operators

The crossover operator exchanges the last third of the genes of two chromosomes to create a new offspring. The typical size of a chromosome in text domains is about one or two thousands genes, and about the first two-thirds of words are almost included in all the chromosomes, although at different places within this fragment. In order to avoid obtaining duplicated genes that furnish no new information, only the last third of chromosomes should be exchanged in the crossover operation. For example, if the parents were:

Chromosome 1: (I, you, he)
Chromosome 2: (we, you, they)

Since the size of chromosome is equal to three and the number of genes of last third is equal to one, then the new offspring would be:

NewChromosome 1: (I, you, they)
NewChromosome 2: (we, you, he)

The mutation operator modifies ten percent of the genes from a randomly selected place *p* in a chromosome by switching them with other possible words from the vocabulary. For example:

Chromosome 1: (I, you, he)
p = 2
Vocabulary = {I, you, he, she, we, you, they}
Size of chromosome: 3
10%*3 = 1 (? 1, by default)
NewChromosome 1: (I, you, we)

The proportion of chromosomes involved in crossover and mutation operations is determined by crossover and mutation probabilities, which are set empirically. In Section 3, the values of these parameters are shown. The results of the application of any genetic operator can produce new chromosomes containing repeated words. Since just the first occurrence of every word within a chromosome will be considered, genetic search can yield

not only an optimal feature set, but also a smaller number of features.

2.2.1.3. Fitness Function

The learner obtains a model for every chromosome of a certain generation. The fitness function of a chromosome is a measurement of the model performance computed on a test sample represented relative to the chromosome. This test sample is a subset of the general test set, and it is composed of relevant, noiseless documents in order to prevent the system from wasting too much time computing the fitness function. The calculation of the fitness function uses about 30% of the documents from the initial test set. All learners use the same test sample, which is then barred from further consideration in order to avoid learning overfitted final categorization models.

Previous research work on wrapper feature selection using genetic algorithms have defined a composed fitness function as a weighted sum of other fitness functions corresponding to different optimization objectives [21]. Because the wrapper approach is very time-consuming, such research has used neural networks as the sole inductive algorithm for evaluating chromosomes and calculating their fitness as an estimate of precision on a test sample. The resulting classifier is more independent of the document sample and shows a lower classification performance.

In HYCLA, the learners deal with a population of fixed size with six chromosomes at most. The initial population is already formed by good feature sets, and the number of generations needed to reach the final feature set is small.

2.2.2. Learning Kernels

When a learner obtains a feature set, the set of training documents is represented relative to that feature set, and then the learner applies its inductive algorithm to learn a classification model. Since there could be four kinds of redundant information in documents, the system can run four learners: the abstract/meta information learner, the reference/link information learner, the contents/plain information learner and the title/url information learner. The selected learning methods embodied in learners are:

- Naïve Bayes [9] for the plain text learner, since the plain text vocabulary is the largest and the noisiest.
- Decision trees [9] for the abstract/meta text and the title/*url* learners. The vocabulary sizes of these types of information are small, and the statistical measurement scores are high. Abstract/meta information is very accurate and contains very little noise. Specifically, the learner runs a C4.5 algorithm [23].
- Rule discovery [5], [6], [7] for the reference/link learner. In hypertext documents, the information contained in links is very rich, because links describe how documents are connected and the web net is formed. The feature set size is the smallest here, and the rules discovered can express all the richness of the information in an understandable manner. The algorithm used is an adaptation to textual domains of a learning method developed earlier by the authors of this paper [4]. This algorithm is based on AQ learning [17] where the seed instance has been replaced by the feature set found by the learner. This difference may reduce the number of expressions candidates to become a general classification rule. The algorithm learns the most general rule for each document category.

2.2.3. Testing

When a learner obtains a classification model, whether the feature set is a tentative one obtained from a certain generation of the genetic algorithm or the optimal one, obtained from the last generation, the model is applied to a test set, and several predictive measurements can then be calculated: *recall* or percentage of documents for a category correctly classified, *precision* or percentage of predicted documents for a category correctly classified, and *F-measure*, which can be viewed as a function made up of the recall and precision measurements. The value of *F-measure* is the fitness value used by the genetic algorithm for chromosomes representing tentative feature sets.

2.3. Integrated Prediction

In order to classify a document, the different kinds of information belonging to the document are represented according to the learned vocabularies of every learner, and then every learner applies its model to make a prediction. Abstract/meta and reference/link texts usually give accurate information about the category of a document. However, there are many documents that lack both these kinds of information, and the system then has to rely on the prediction made by the plain text and url learners.

There are two options for obtaining the final classification prediction of a document:

- To take the model with the best performance results for classification in the testing stage (i.e. F-measure) as the optimal final solution.
- To take a combination of the models as the final solution. The combination can be determined as an average or a weighted sum of the individual predictions. The weights of individual learners can be any of the computed performance measurements or can be set by a genetic algorithm [14].

HYCLA performs a weighted integration of the individual predictions, and it determines the weight of each learner together with that of the other learners by using a genetic algorithm. This genetic integration contributes to improve the results in classification performance of imbalanced thematic categories without some type of information.

2.3.1. Genetic Integration

The genes of a chromosome represent the weights, between 0 and 1, of the predictions made by the different learners. Chromosome length matches the number of learners involved in the problem. The initial population is made up of chromosomes whose genes take values from the set [0.0, 0.2, 0.4, 0.6, 0.8, 1], allowing all possible combinations of these values, and an additional chromosome whose genes are the values of F-measure obtained by each learner in the testing stage.

The crossover operator allows the genes of two parent chromosomes, taken from a randomly selected place, to be exchanged. The mutation operator increases or reduces the weight of a randomly selected gene by a quantity between [-0.10...0.10].

The fitness function evaluates every chromosome on a labeled test set by combining chromosome weights. Each learner predicts a category for a document with a weight equal to the gene in the chromosome that represents the learner. When several learners

predict the same category, the average of their weights is calculated. The final predicted category for a document will be the one predicted by the learner or learners with the highest weight. For example, for the following chromosome:

Chromosome: (0.7, 0.8, 0.85, 0.5, 0.97)

where (Chromosome[i]=Weight [Learner i])

If the predictions of the learners were:

Learners 1,3,5: Prediction = Category 1;

Average Weight = 0.84

Learner 2 : Prediction = Category 3;

Weight = 0.8

Learner 4 : Prediction = Category 2;

Weight = 0.5

The highest weight is 0.84, and so the resulting prediction assigns the document to Category 1.

The fitness function value of a chromosome is the value of F-measure achieved by the chromosome for the full test set of documents. The stop criterion of the genetic search could be a threshold value of the fitness function, i.e. a classification precision of 97%, or a certain number of generations.

The computational cost of this genetic search is very low, since the classification of test documents has been performed by the learners in the model testing stage.

3. EMPIRICAL EVALUATION

HYCLA has been evaluated on three text collections. These collections are described below, followed by a review of the experimental settings and results.

Reuters-21578 is a collection of newswire article texts that appeared in 1987. The entire collection has 21,578 texts belonging to 135 topic categories. In order to evaluate the performance of HYCLA, the sample taken into account is composed of categories with more than 100 examples. The selected example set has a size of 12,066 documents belonging to 23 different categories. These documents were arranged into three

Examples from Reuters-21578 Corpus	Training1	Test11	Test12	Total
1. ACQ	1,200	500	351	2,402
2. BOP	59	25	17	118
3. COFFEE	75	32	21	150
4. CORN	126	50	39	225
5. CPI	58	25	17	117
6. CRUDE	296	140	78	592
7. DLR	122	55	33	244
8. EARN	1,860	800	530	3,721
9. GNP	82	33	25	166
10. GOLD	69	25	22	139
11. GRAIN	311	145	83	622
12. INTEREST	245	110	73	491
13. LIVESTOCK	56	20	18	113
14. MONEY-FX	406	190	108	812
15. MONEY-SUPPLY	101	45	23	202
16. NAT-GAS	56	34	21	132
17. OILSEED	96	41	23	193
18. SHIP	142	62	39	283
19. SOYBEAN	58	25	17	117
20. SUGAR	94	41	26	188
21. TRADE	282	125	78	564
22. VEG-OIL	67	26	21	136
23. WHEAT	153	65	45	309
Total	6,014	2,614	1,708	12,066

Table 1. Distribution into categories and arrangement into training and test sets of the articles selected from Reuters-21578.

disjoint subsets (see Table 1): a training set, *Training1*, with 6,014 documents, and two test sets, *Test11*, with 2,614 documents, and *Test12*, with 1,708 documents.

Another collection of 7,161 text documents was collected by a program that automatically downloads web documents. The documents belong to three different categories, and were arranged into three disjoint subsets (see Table 2): a training set, *Training1*, with 5,008 documents, and three test sets, *Test11*, *Test12* and *Test13*, with 1,416, 346 and 391 documents, respectively. The "NOISE" category is composed of error pages and randomly downloaded pages.

downloaded from the web contains documents that lack some part of text information.

A Naïve Bayes classifier was trained on both collections using *Training1*. Test set *Test12* was used by the genetic algorithm to evaluate the fitness function, and test set *Test11* was used to evaluate the classification accuracy measurements of the learned models.

The final size of the vocabulary of *Training1* on Reuters-21578, after removing the words from a stop list, was 16,806. The value of k_v chosen for running the feature selection methods was 5,041.

Examples downloaded from WWW	Training1	Test11	Test12	Test13	Total
GAMBLING	1,978	560	117	166	2,821
GAMES	1,398	404	115	124	2,041
MUSIC	1,437	311	114	101	1,963
NOISE	195	141	0	0	336
Total	5,008	1,416	346	391	7,161

Table 2. Distribution into categories and arrangement into training and test sets of documents downloaded from Internet.

The third collection is composed of 2,442 documents belonging to five domains defined from the Yahoo Directory. These documents were arranged into three disjoint subsets (see Table 3): *Training1*, with 1,121 documents, and two test sets, *Test11* and *Test12*, with 561 and 561 documents, respectively.

In the HTML collection, only the words contained in the HTML tag "<META>" were taken into account. The vocabulary size, after removing stop-list words, was 9,364, and the value of k_v was 3,000.

Examples downloaded from Yahoo Hierarchy	Training1	Test11	Test12	Total
ARTS	272	136	136	544
COMPUTERS	170	85	85	340
EDUCATION	136	68	68	271
ENTERTAINMENT	210	105	105	420
REFERENCES	333	167	167	667
1. Total	1,121	561	561	2,242

Table 3. Distribution into categories and arrangement into training and test sets of documents downloaded from Internet.

3.1. Feature Selection Methods

The first kind of experiment allowed the classification performance of several feature selection methods to be compared and showed the improvement achieved by evolutionary selection method used by HYCLA. The statistical methods used were: information gain (I.G.), document frequency (D.F.), chi square (CHI^2), crossover entropy (C.E.), mutual information (M.I.) and odds-ratio (O.R.). The values of crossover and mutation probabilities used for the evolutionary feature selection were both 0.4.

This experiment was performed on the two first imbalanced collections: Reuters-21578 collection contains categories with many documents while others do not and HTML collection

The performance of each method is reported below, in Table 4 for the Reuters texts and in Table 5 for the downloaded web pages. The numerical values *Pr*, *Rc* and *F* represent precision, recall and F-measure (F = (2 * precision * recall) / (precision + recall)) normalized between zero and one, respectively. In Table 4, the columns show these values obtained by each feature selection method in each category. The last row presents the macro averaged values. In Table 5, the rows show the precision, recall and F-measure values obtained by each feature selection in each category. The last column shows the macro averaged values. The values of the evolutionary feature selection method are the average values from running the genetic algorithm five times.

In both tables boldface indicates the best values in each category. The results indicate that each statistical feature selection method behaves the best only in certain categories. The genetic feature selection method yields the best average F-measure value in all categories on both collections. This fact reflects that the genetic feature selection method is more independent of the distribution of the examples into categories than the other selection methods, and therefore more robust for handling imbalanced data. In the categories where the genetic method does not give the best performance, it does yield the second or third best value. Moreover, the best average F-measure implies the best ratio between precision and recall. The other feature selection methods have a certain leaning towards one or the other of these two measurements. The experiments show that a significant departure from the approaches that utilize universal feature selection yields better results.

	I.G.			D.F			CHI^2			C.E.			M.I.			O.R.			G.A. *(average of five runs)*		
	Pr	*Re*	*F*	*Pr*	*Re*	*F*	*Pr*	*Re*	*F*	*Pr*	*Re*	*F*	*Pr*	*Re*	*F*	*Pr*	*Re*	*F*	*Pr*	*Re*	*F*
1	69.2	**99.2**	0.81	69.0	**99.2**	0.81	67.8	87	0.76	69.2	**99.2**	0.81	**76.7**	73.4	0.75	62.7	83.6	0.71	71.2	**99.2**	**0.82**
2	**100**	32	0.48	**100**	32	0.48	33.3	40	**0.54**	**100**	32	0.48	**100**	4	0.07	**100**	32	0.48	**100**	32	0.48
3	**100**	25	0.40	**100**	21.8	0.35	77.7	21.8	0.34	**100**	21.8	0.35	81.0	**93.7**	**0.86**	**100**	31.2	0.47	**100**	25	0.40
4	**100**	26	0.41	**100**	26	0.41	81.2	**52**	**0.63**	**100**	26	0.41	28	14	0.18	**100**	26	0.41	**100**	26	0.41
5	**100**	20	0.33	**100**	20	0.33	92.8	52	**0.66**	**100**	20	0.33	80	48	0.60	**100**	20	0.33	**100**	20	0.33
6	69.7	59.2	0.64	69.1	59.2	0.63	**80.2**	43.5	0.56	69.7	59.2	0.64	56.8	**82.8**	0.67	70.2	67.3	**0.68**	70.9	59.2	0.64
7	**100**	40	0.57	**100**	40	0.57	70.3	34.5	0.46	**100**	38.1	0.55	88.3	**69.0**	0.77	**100**	40	0.57	**100**	40	0.57
8	96.3	91.7	0.93	96.3	**92**	**0.94**	65.5	91.7	0.76	96.3	91.7	0.93	84.9	85.1	0.85	**97.1**	89.6	0.93	96.3	91.8	**0.94**
9	**100**	27.2	0.42	**100**	27.2	0.42	87.5	42.4	**0.57**	**100**	27.2	0.42	50	**48.4**	0.49	100	27.2	0.42	**100**	27.2	0.42
10	**100**	12	0.21	**100**	12	0.21	**100**	12	0.21	**100**	12	0.21	76.9	**80**	**0.78**	92.8	52	0.66	**100**	28	0.43
11	42.6	87.5	0.57	42.0	87.5	0.56	**70.6**	56.5	**0.62**	42.6	87.5	0.57	34.7	66.2	0.45	44.3	**89.1**	0.59	41.6	88.2	0.56
12	**100**	34.5	0.51	**100**	35.4	0.52	90.5	**60.9**	**0.72**	**100**	34.5	0.51	75	49.0	0.59	**100**	34.5	0.51*	**100**	34.5	0.51
13	**100**	15	0.26	**100**	15	0.26	70	35	0.46	**100**	15	0.26	**100**	15	0.26	**100**	10	0.18	81.8	**45**	**0.58**
14	53.2	**95.7**	0.68	53.2	**95.7**	0.68	**80.1**	67.8	**0.73**	52.9	94.7	0.67	55.6	60	0.57	53.2	**95.7**	0.68	53.3	**95.7**	0.68
15	**100**	42.2	0.59	**100**	42.2	0.59	91.1	**68.8**	**0.78**	**100**	42.2	0.59	82.1	51.1	0.63	99.2	45	0.61	**100**	42.2	0.59
16	**100**	14.7	0.25	**100**	17.6	0.30	83.3	**29.4**	**0.43**	**100**	14.7	0.25	50	2.94	0.05	**100**	17.6	0.3	**100**	17.6	0.30
17	**100**	19.5	0.32	**100**	17.0	0.29	86.2	**60.9**	**0.71**	**100**	12.1	0.21	33.3	9.75	0.15	99.1	50.1	0.66	**100**	19.5	0.32
18	70.5	19.3	0.30	**71.4**	16.1	0.26	69.2	14.5	0.24	70.5	19.3	0.30	55.5	**64.5**	**0.59**	52	40	0.45	64.2	43.5	0.51
19	**100**	20	0.33	**100**	20	0.33	87.5	56	**0.68**	**100**	20	0.33	7.41	4	0.05	**100**	20	0.33	**100**	20	0.33
20	**100**	19.5	0.32	**100**	19.5	0.32	**100**	17.0	0.29	**100**	19.5	0.32	82.1	**56.0**	0.66	81.2	31.7	0.45	**100**	46.3	0.63
21	54.1	84	**0.65**	54.6	**84.8**	0.66	**78.8**	44.8	0.57	54.1	84	**0.65**	46.6	76.8	0.58	54.7	36.8	0.44	53.5	84	0.65
22	**100**	15.3	0.26	**100**	15.3	0.26	38.4	19.2	0.25	**100**	23.0	0.37	**100**	23.0	0.37	**100**	23.0	0.37	90	**34.6**	**0.50**
23	**100**	9.23	0.16	**100**	9.23	0.16	86	**66.1**	**0.74**	**100**	9.23	0.16	17.2	7.69	0.10	60	11.5	0.19	**100**	9.23	0.16
Avg	**89.3**	39.5	0.54	**89.3**	39.5	0.54	79.9	46.7	0.58	89.3	39.2	0.54	63.5	**47.1**	0.54	83.5	42.3	0.56	87.9	44.7	**0.59**

Table 4. Comparative performance measurements among feature selection methods obtained on the Reuters-21578 collection.

Feature Selection Results	GAMBLING			GAMES			MUSIC			AVERAGE		
	Pr.	*Rc.*	*F*	*Pr.*	*Rc.*	*F*	*Pr.*	*Rc.*	*F*	*Pr.*	*Rc.*	*F*
I.G.	52%	**99%**	0.68	**99%**	37%	0.53	**99%**	70%	0.82	83%	69%	0.75
D.F.	51%	99%	0.67	**99%**	36%	0.52	**99%**	69%	0.81	83%	68%	0.74
CHI^2	54%	88%	0.67	80%	52%	0.63	98%	**78%**	**0.86**	77%	72%	0.74
C.E.	52%	**99%**	0.68	**99%**	47%	0.64	**99%**	70%	0.82	83%	72%	0.77
M.I.	55%	90%	0.68	81%	**53%**	0.64	96%	71%	0.81	77%	71%	0.73
O.R.	48%	98%	0.65	87%	31%	0.45	**99%**	75%	0.85	78%	68%	0.72
G.A. *(average of five runs)*	**56%**	**99%**	**0.71**	**99%**	50%	**0.66**	**99%**	72%	0.83	**84%**	**73%**	**0.78**

Table 5. Comparative performance measurements among feature selection methods obtained on HTML collection.

3.2. Integration of Predictions

The second kind of experiment was set up to compare the performance of the predictions of every individual learner and the genetic combination of these predictions. A comparison of genetic combination of predictions and a voting combination is also reported. The voting combination proposes to assign the category predicted by a majority of individual learners to an incoming document.

There are four learners for the four different types of information taken into account in HTML documents, *url* text, meta-text, plain text and hyperlink text.

The experiments were performed on the free collection downloaded from the web shown in Table 2 and the Yahoo collection shown in Table 3. Table 6 and Table 8 present the size of the initial vocabularies of *Training1* after removing stop-list words and the value of k_v for each type of information.

program as well as the non-informative arguments inherent to the document.

The values of crossover and mutation probabilities used for the evolutionary integration were 0.7 and 0.4, respectively.

Table 7 and Table 9 show the precision, recall and F-measure values of every learner and of the genetic and voting combination in every category. Last rows indicate the average values. Boldface indicates the best performance values in every category. The

Vocabulary size in Training1 from WWW	**URL** Vocabulary	**META** Vocabulary	**TEXT** Vocabulary	**LINK** Vocabulary
Total size	5,164	9,364	+30,000	+ 30,000
Feature selection size	1,550	2,810	10,000	10,000

Table 6. Initial vocabulary size s and k_v for each type of information in downloaded collection.

WWW Collection	**URL**			**META**			**TEXT**			**LINKS**		
	Pr	*Rc*	*F*	*Pr*	*Rc*	*F*	*Pr*	*Rc*	*F*	*Pr*	*Rc*	*F*
GAMBLING	**100**	32.96	0.495	77.9	98.72	0.870	76.2	98.4	0.858	83.12	96.96	0.845
GAMES	53.62	**96.88**	0.690	94.6	74	0.830	93.59	74.66	0.830	92.56	80.22	**0.859**
MUSIC	68.51	79.76	0.737	95.74	79.17	**0.866**	**95.9**	68.62	0.800	89.65	76.24	0.824
Average	74.04	69.86	0.718	89.41	83.96	0.865	88.56	80.56	0.843	88.44	**84.47**	0.864
	COMBINATION *(avg. of five runs)*			**VOTING**								
	Pr	*Rc*	*F*	*Pr*	*Rc*	*F*						
GAMBLING	79.08	**99.2**	**0.880**	80.3	96.8	0.877						
GAMES	**96.28**	74.88	0.842	92.6	73.1	0.817						
MUSIC	95.58	76.24	0.848	87.6	**80.5**	0.839						
Average	**90.31**	83.44	**0.867**	86.83	83.46	0.851						

Table 7. Comparative performance measures of independent parts, genetic combination and voting combination classifiers on downloaded HTML collection.

free collection, *Test 13* is the set used to calculate the fitness function in the genetic combination of predictions, and *Test11* is the set used to calculate the performance measurements.

On the Yahoo dataset, the *url* text of documents has not been included since it gives no relevant information. The access to

genetic performance values are the average values found by running the genetic algorithm five times.

This experiment shows that the average F-measure of genetic integration is the best result. Individual learners obtain good results only in certain categories. The voting combination of

Vocabulary size in Training1 from Yahoo Hierarchy	**META** Vocabulary	**TEXT** Vocabulary	**LINK** Vocabulary
Total size	8,946	26,000	+ 30,000
Feature selection size	3,000	8,600	10,000

Table 8. Initial vocabulary size s and k_v for each type of information in Yahoo collection.

these web documents was performed through links in the Yahoo Directory. Every link activates a specific url that runs a program leading the Internet browser to such web documents. The *url* text of each downloaded web document contains the url of the

learner predictions obtains very poor results in all categories. The genetic combination of learner predictions behaves better and more smoothly than individual predictions and voting combination in all categories.

	META			TEXT			LINKS		
	Pr	*Rc*	*F*	*Pr*	*Rc*	*F*	*Pr*	*Rc*	*F*
ARTS	75.57	**72.79**	**0.741**	**93.15**	50	0.650	90	39.70	0.551
COMPUTERS	**85.71**	56.47	**0.680**	52.34	**78.82**	0.629	62.22	65.88	0.639
EDUCATION	**95.34**	**60.29**	**0.738**	94.28	48.52	0.640	90	26.47	0.409
ENTERNTAINMENT	95.23	57.14	0.714	**95.78**	**86.66**	**0.909**	97.5	74.28	0.843
REFERENCES	54.10	86.82	0.666	**63.91**	**88.02**	**0.740**	48.23	89.82	0.627
Average (macro)	**81.19**	66.70	0.732	79.89	**70.40**	**0.748**	77.59	59.23	0.671

	COMBINATION *(average of five runs)*			**VOTING**		
	Pr	*Rc*	*F*	*Pr*	*Rc*	*F*
ARTS	85.50	**43.38**	**0.575**	**91.80**	41.17	0.568
COMPUTERS	**46.61**	**64.70**	**0.541**	45	63.52	0.526
EDUCATION	**100**	**35.29**	**0.521**	91.66	16.17	0.274
ENTERNTAINMENT	**93.68**	**84.76**	**0.889**	92.85	74.28	0.825
REFERENCES	**52**	**85.62**	**0.647**	51.30	82.63	0.633
Average (macro)	**75.55**	**62.75**	**0.685**	74.52	55.55	0.636

Table 9. Comparative performance measures of independent parts, genetic combination and voting combination classifiers on Yahoo collection.

4. CONCLUSIONS

The architecture presented in this paper is a combination of a variable number of learners. Learners may be added or removed depending on the specific text categorization task. This modularity makes the system adaptable to any particular context.

The genetic feature selection method takes advantage of each statistical selection method used. This method works quite well for all categories, regardless of the distribution of the documents in the training sample. Moreover, statistical feature selection methods display text-domain dependence, and the evolutionary method makes this dependence smoother.

The division of HTML documents into four types of text has shown that some words have a greater importance in a certain piece of text than in the full text with no partition. The application of different learners to each type of information allows the system to be independent of text domain without loss of accuracy. The genetic integration of the predictions of the learners yields good results in classification performance.

5. FUTURE WORK

Currently work in this area is mainly focused on the design and development of a genetic algorithm devoted to discovering the classification models of different categories of documents. The entire text classification task could be carried out by a genetic algorithm alone. Simplicity, uniformity and intelligibility would be the main features of the resulting categorization system.

Classifying a new document would mean measuring the distance between the suitably represented document and the chromosome or model being evaluated. The definition of the distance measurement and genetic operators are the key points of this research. The individuals that are revealed as the best would be the optimal classification models.

6. ACKNOWLEDGMENT

Part of this work was supported by the Spanish Ministry of Science and Technology under project FIT-070000-2001-193 and by Optenet, S.A.

7. REFERENCES

[1] Attardi G., Gulli A., Sebastiani F.: Automatic Web Page Categorization by Link and Content Analysis. Proceedings of THAI'99, European Symposium on Telematics, Hypermedia and Artificial Intelligence. Varese (1999) 105-119.

[2] Brank, J., Groblenik, M., Milic-Frayling, N., Mladenic, D.: Interaction of Feature Selection Methods and Linear Classification Models. Proceedings of the Nineteenth International Conference on Machine Learning (ICML'02). Sydney, Australia (2002).

[3] Castillo, Mª. D. del, Gasós, J., García-Alegre, M.C.: Genetic Processing of the Sensorial Information. Sensors & Actuators A, 37-38 (1993) 255-259.

[4] Castillo, Mª. D. del, Barrios, L. J.: Knowledge Acquisition from Batch Semiconductor Manufacturing Data. Intelligent Data Analysis IDA, 3, Elsevier Science Inc. (1999) 399-408.

[5] Castillo, Mª. D. del, Sesmero, P.: Perception and Representation in a Multistrategy Learning Process. Proceedings of Learning'00. Madrid (2000).

[6] Cohen, W.: Text categorization and relational learning. Proceedings of the Twelfth International Conference on Machine Learning. Lake Tahoe, California (1995) 124-132.

[7] Craven, M., DiPasquo, D., Freitag, D., McCallum, A., Mitchell, T., Nigam, K., Slattery, S.: Learning to Construct Knowledge Bases from the World Wide Web. Artificial Intelligence,118(1-2) (2000) 69-113.

[8] Doan, A., Domingos, P., Halevy, A.: Learning to Match the Schemas of Data Sources: A Multistrategy Approach. Machine Learning, Vol. 50 (2003) 279-301.

[9] Dumais, S. T., Platt, J., Heckerman, D., and Sahami, M.: Inductive Learning Algorithms and Representation for Text Categorization. In CIKM-98: Proceedings of the Seventh International Conference on Information and Knowledge Management (1998) 148-155.

[10] Freitag, D.: Multistrategy Learning for Information Extraction. Proceedings of the 15th International Conference on Machine Learning (1998) 161-169.

[11] Goldberg, D.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison Wesley (1989).

[12] Grobelnik, M., Mladenic, D.: Efficient Text Categorization. Proceedings of the ECML-98 Text Mining Workshop (1998).

[13] John, G.H., Kohavi, R., Pfleger, K.: Irrelevant Features and the Subset Selection Problems. Proceedings of the 11th International Conference on Machine Learning (1994).

[14] Langdon, W. B., Buxton, B. F.: Genetic Programming for Combining Classifiers. Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001) (2001) 66-73.

[15] Lewis, D.: Feature selection and feature extraction for text categorization. Proceedings of Speech and Natural Language Workshop. Defense Advanced Research Projects Agency, Morgan Kaufmann, February (1992) 212-217.

[16] Lewis, D., Ringuette, M.: A Comparison of Two Learning Algorithms for Text Categorization. Symposium on Document Analysis and IR, ISRI, April 11-13, Las Vegas (1994) 81-93.

[17] Michalski, R.S., Carbonell J.G., Mitchell T.M.: A theory and methodology of inductive learning. Machine Learning: An Artificial Intelligence Approach. Springer-Verlag (1983).

[18] Mladenic, D.: Feature Subset Selection in Text-Learning. European Conference on Machine Learning (1998) 95-100.

[19] Mladenic, D., Grobelnik, M.: Feature selection for classification based on text hierarchy. Working notes of Learning from Text and the Web, Conference on Automated Learning and Discovery CONALD-98 (1998).

[20] Mladenic, D., Grobelnik, M.: Feature selection for unbalanced class distribution and Naïve Bayes. Proceedings of the 16th International Conference on Machine Learning (ICML'99) (1999) 258-267.

[21] Oliveira, L. S.: Feature Selection Using Multi-Objective Genetic Algorithms for Hand-written Digit Recognition, ICPR (2002).

[22] Porter, M.F.: An algorithm for suffix stripping. Program, 14(3) (1980) 130-137.

[23] Quinlan J. R.: C4.5: Programs for Machine Learning. San Mateo, CA: Morgan Kaufmann (1993).

[24] Sebastiani, F.: Machine Learning in Automated Text Categorization. ACM Computing Surveys, Vol. 34, Number 1 (2002) 1-47.

[25] Yang, Y., Pedersen, J.P.: A Comparative Study on Feature Selection in Text Categorization. Proceedings of the Fourteenth International Conference on Machine Learning (ICML'97) (1997) 412-420.

[26] Yang, J. and Honavar, V.: Feature subset selection using a genetic algorithm. IEEE Intelligent Systems and their Applications. 13(2) (1998) 44-49.