Multi-label testing for CO^{2}RBFN. A first approach to the problem transformation methodology for multi-label classification

A. J. Rivera, F. Charte, M. D. Pérez-Godoy and María Jose del Jesus

Dep. of Computer Science, University of Jaén, Jaén, Spain
e-mail:arivera@ujaen.es http://simidat.ujaen.es

Abstract. Whereas that in traditional classification an instance of the data set is only associated with one class, in multi-label classification this instance can be associated with more than one class or label. Examples of applications, in this growing area, are: text categorization, functional genomics or semantic association of audio or video. One way to address these applications is the Problem Transformation methodology that transforms the multi-label problem into one single-label classification problem, in order to apply traditional classification methods. The aim of this contribution is to test the performance of CO^{2}RBFN, a cooperative-competitive evolutionary model for the design of RBFNs, in a multi-label environment, using the problem transformation methodology. The results obtained by CO^{2}RBFN, and by other classical data mining methods, show an irregular behavior depending on the problem transformation, data set or measure used.

Keywords: Multi-label Classification, RBFNs, Problem Transformation.

1 Introduction

Recently, applications where an instance of the data set is associated with several labels or classes are appearing. For example in text categorization, each document can be classified to different predefined topics, such as *education* and *health*, a movie may belong to the classes *action* and *thriller*, or a song can be categorized as *rock* and *pop*. These data sets are called multi-label data set and the related classification task is called multi-label classification [11][3].

The first applications [11] in this area dealt with text categorization problems but other examples are: functional genomics, semantic association of images, scene classification, medical diagnosis or directed marketing.

The different approaches that address multi-label classification can be categorized into two groups: Problem Transformation and Algorithm Adaptation. The first group of algorithms transforms the multi-label problem into one single-label classification problem. In the second group, classical algorithms are adapted to handle multi-label data directly.

Radial Basis Function Networks (RBFNs) are one of the most important Artificial Neural Network (ANN) paradigms in the machine learning field. An RBFN is a feed-forward ANN with a single layer of hidden units, called radial basis functions (RBFs) [1]. The overall efficiency of RBFNs has been proved in many areas [2] like pattern classification, function approximation and time series prediction.

An important paradigm for RBFN design is the Evolutionary Computation [6]. There are different proposals in this area with different scheme representations: Pittsburgh [8], where each individual is a whole RBFN, and cooperative-competitive [12], where an individual represents an RBF.

Authors have developed an algorithm for the cooperative-competitive design of Radial Basis Functions Networks, CO^2RBFN [10], that has been successfully used in classical and imbalanced classification.

The purpose of the present paper is to test CO^2RBFN in multi-label classification, exploring this new field. For this initial approach and based on the first group of mentioned techniques, multi-label data sets are transformed into single-label data sets. Results obtained are compared with other traditional techniques in data mining.

The text is organized as follows. In Section 2, multi-label classification and the solutions provided for them in the specialized bibliography are described. The cooperative-competitive evolutionary model for the design of RBFNs applied to classification problems, CO^2RBFN, is described in Section 3. The analysis of the experiments and the conclusions are shown in Sections 4 and 5.

2 Multi-label classification

Classification is one of the most important tasks in data mining. In a classification environment, a mapping from an input space X^n to a finite set of classes L with $L = \{l_1, l_2, ..., l_k\}$, must be established. Considering a training set D with p patterns or instances:

$$D = \{(\boldsymbol{x}_u, l_u) | \boldsymbol{x}_u \in X^n, l_u \in L, u = 1, \ldots, p\} \quad (1)$$

where $\boldsymbol{x}_u$ is the feature vector and l_u is the class it belongs to. When $|L| = 2$ the classifier is said binary. If $|L| > 2$ a multi-class classifier is needed. In any case, each instance is only associated with one of the classes.

However, there is an important number of problems where each instance can be simultaneously associated with a subset of classes or labels $Y \subseteq L$. These problems are known as multi-label classification problems. Even binary classification and multi-class classification can be seen as special cases of multi-label problems where the number of labels assigned to each instance is 1.

As mentioned, there are two mainly ways to address multi-label classification problem [11]: Problem Transformation and Algorithm Adaptation approaches. In the problem transformation (algorithm independent) way, the original problem is transformed into a set of single-label problems. The most popular of these transformations are:

– Label powerset (LP): this method considers as a single label, the subset of labels associate with each instance of the data set. Drawbacks of this method, that complicate the learning process, are: the obtained data set can have a large number of classes, and some of these classes can be associated with few examples.
– Binary relevance (BR): this method, based on one-against-all techniques, creates a new data set for each label of the original data set. Thus, for example, in the $i-th$ data set, each instance associated with the label i is labeled as positive and the rest of instances are labeled as negatives. As drawback, this method may not be able of handling correlations among labels.

Despite their possible drawbacks, BR and LP can achieve reasonably good results and we will use it in our experimentation.

On the other hand, algorithm adaptation approaches modify existing algorithms to manage multi-label data. For example, ML-kNN [15], a modification of the well-known kNN algorithm, uses prior and posterior probabilities for the frequency of labels within the k nearest neighbors, in order to determine the label set of a test instance. In [4] C4.5 algorithm was adapted modifying the calculation of its formula of entropy in order to manage multi-label data. BP-MLL [16] introduces a new error function, in the Backpropagation algorithm, in order to take into account multiple labels. A modification of the SVM algorithm that minimizes the ranking loss measure is proposed in [5]. ML-RBF [14] uses a clustering-based analysis for each label in order to place the neurons of the net, existing an output in the RBFN for each label.

3 CO^{2}RBFN: an evolutionary cooperative-competitive hybrid algorithm for RBFN design

CO^{2}RBFN [10], is an evolutionary cooperative-competitive hybrid algorithm for the design of RBFNs. In this algorithm each individual of the population represents, with a real representation, an RBF and the entire population is responsible for the final solution.

The individuals cooperate towards a definitive solution, but they must also compete for survival. In this environment, in which the solution depends on the behaviour of many components, the fitness of each individual is known as credit assignment.

In order to measure the credit assignment of an individual, three factors have been proposed: the RBF contribution to the network output, the error in the basis function radius, and the degree of overlapping among RBFs.

The application of the operators is determined by a Fuzzy Rule-Based System. The inputs of this system are the three parameters used for credit assignment and the outputs are the operators' application probability.

The main steps of CO^{2}RBFN, explained in the following subsections, are shown in the pseudocode, in Figure 1.

```
1. Initialize RBFN
2. Train RBFN
3. Evaluate RBFs
4. Apply operators to RBFs
5. Substitute the eliminated RBFs
6. Select the best RBFs
7. If the stop condition is not verified go to step 2
```

Fig. 1. Main steps of CO^2RBFN

RBFN initialization. To define the initial network a specified number m of neurons (i.e. the size of population) is randomly allocated among the different patterns of the training set. The RBF widths, d_i, will be set to half the average distance between the centres. Finally, the RBF weights, w_{ij}, are set to zero.

RBFN training. The Least Mean Square algorithm [13] has been used to calculate the RBF weights.

RBF evaluation.A credit assignment mechanism is required in order to evaluate the role of each RBF ϕ_i in the cooperative-competitive environment. For an RBF, three parameters, a_i ,e_i ,o_i are defined:

– The contribution, a_i, of the RBF ϕ_i, is determined by considering the weight, w_i, and the number of patterns of the training set inside its width, pi_i:

$$a_i = \begin{cases} |w_i| & if \quad pi_i > q \\ |w_i| * (pi_i/q) & otherwise \end{cases} \tag{2}$$

where q is the average of the pi_i values minus the standard deviation of the pi_i values.

– The error measure, e_i, for each RBF ϕ_i, is obtained by counting the wrongly classified patterns inside its radius:

$$e_i = \frac{pibc_i}{pi_i} \tag{3}$$

where $pibc_i$ and pi_i are the number of wrongly classified patterns and the number of all patterns inside the RBF width respectively.

– The overlapping of the RBF ϕ_i and the other RBFs is quantified by using the parameter o_i. This parameter is computed by taking into account the fitness sharing methodology [6], whose aim is to maintain the diversity in the population.

Applying operators to RBFs. In CO^2RBFN four operators have been defined in order to be applied to the RBFs:

– Operator Remove: eliminates an RBF.
– Operator Random Mutation: modifies the centre and width of an RBF in a random percentage of the old width.
– Operator Biased Mutation: modifies, using local information, the RBF trying to locate it in the centre of the cluster of the represented class.

– Operator Null: in this case all the parameters of the RBF are maintained.

The operators are applied to the whole population of RBFs. The probability for choosing an operator is determined by means of a Mandani-type fuzzy rule based system [9] which represents expert knowledge about the operator application in order to obtain a simple and accurate RBFN. The inputs of this system

Table 1. Fuzzy rule base representing expert knowledge in the design of RBFNs

Antecedents				Consequents				Antecedents				Consequents			
	v_a	v_e	v_o	p_{remove}	p_{rm}	p_{bm}	p_{null}		v_a	v_e	v_o	p_{remove}	p_{rm}	p_{bm}	p_{null}
R1	L			M-H	M-H	L	L	R6		H		M-H	M-H	L	L
R2	M			M-L	M-H	M-L	M-L	R7			L	L	M-H	M-H	M-H
R3	H			L	M-H	M-H	M-H	R8			M	M-L	M-H	M-L	M-L
R4		L		L	M-H	M-H	M-H	R9			H	M-H	M-H	L	L
R5		M		M-L	M-H	M-L	M-L								

are parameters a_i, e_i and o_i used for defining the credit assignment of the RBF ϕ_i. These inputs are considered as linguistic variables va_i, ve_i and vo_i. The outputs, p_{remove}, p_{rm}, p_{bm} and p_{null}, represent the probability of applying Remove, Random Mutation, Biased Mutation and Null operators, respectively. Table 1 shows the rule base used to relate the described antecedents and consequents.

Introduction of new RBFs. In this step, the eliminated RBFs are substituted by new RBFs. The new RBF is located in the centre of the area with maximum error or in a randomly chosen pattern with a probability of 0.5 respectively.

Replacement strategy. The replacement scheme determines which new RBFs (obtained before the mutation) will be included in the new population. To do so, the role of the mutated RBF in the net is compared with the original one to determine the RBF with the best behaviour in order to include it in the population.

4 Experimentation

The objective of this paper is to test our present evolutionary cooperative-competitive algorithm for RBFN design, CO^{2}RBFN, in the new multi-label classification field and taking into account other typical data mining methods. With the conclusions obtained, we can draw lines for future developments.

With this purpose in mind, we have used the multi-label data mining software and repository Mulan (http://mulan.sourceforge.net/index.html). In this site you can find different multi-label methods, tools and data sets as well as the possibility of using classical Weka learning methods [7].

In order to test CO^{2}RBFN the data sets Emotions and Scene have been chosen. In Emotions a piece of music must be classified in more than one class and in Scene an image may belong to multiple semantic classes. Emotions has 593 instances, 72 numeric attributes and 6 labels. Scene has 2407 instances, 294 numeric attributes and 6 labels. As first conclusion, consequence of reviewing multi-label data sets, the high dimensionality of these ones must be highlighted.

Typical data-mining methods have been chosen for comparisons, specifically: C4.5, KNN, Naive Bayes, MLP, PART, RBFN and SVM. Their implementations and references can be found in Weka [7]. These methods have been run with their default parameters. For CO^2RBFN the iterations of the main loop have been established to 100 and the number of neurons in the range between 10 and 20.

To run CO^2RBFN and the other classical data mining techniques with the above data sets, we use the problem transformation methodology and concretely the popular Binary Relevance and Label Powerset techniques. In this way, both Emotions and Scene have been transformed with BR and LP.

General experimentation parameters are ten-fold cross validation and 3 repetitions for obtaining the means values of the tables of results. The measures used in the results are the ones returned by Mulan software and are described in [11]. For the measure Hamming Loss the lower value, the better and for the other the higher value, the better. In bold the best result.

In Table 2 the experimentation results for BR transformation and the two data sets are shown. Table 3 shows the results for the LP transformation.

Table 2. Experimentation Results with Binary Relevance transformation

Data set Emotion								
	C4.5	CO^2RBFN	KNN	MLP	Naive Bayes	PART	RBFN	SVM
Hamming Loss	0.247	**0.204**	0.235	0.215	0.252	0.257	0.229	0.244
Subset Accuracy	0.184	**0.270**	0.268	**0.270**	0.206	0.157	0.213	0.180
Example-Based Recall	0.599	0.612	0.626	0.646	**0.773**	0.614	0.630	0.441
Example-Based Accuracy	0.462	0.514	0.514	0.525	**0.529**	0.456	0.494	0.391
Data set Scene								
	C4.5	CO^2RBFN	KNN	MLP	Naive Bayes	PART	RBFN	SVM
Hamming Loss	0.137	0.141	0.111	**0.100**	0.242	0.119	0.139	0.126
Subset Accuracy	0.427	0.365	**0.629**	0.566	0.169	0.477	0.369	0.306
Example-Based Recall	0.634	0.457	0.693	0.706	**0.858**	0.668	0.484	0.325
Example-Based Accuracy	0.535	0.419	**0.674**	0.647	0.453	0.578	0.437	0.323

As can be observed, from the tables of results there is not a method that outperforms the others, neither for BR transformation nor for the LP transformation. Another conclusion, that can be extracted, is that methods have irregular performances depending on the problem transformation, data set or measure used. CO^2RBFN achieves its best results for Emotions data set (independently of the transformation used) outperforming the rest of the methods in four measures. For the BR transformation of Scene, CO^2RBFN achieves re-

Table 3. Experimentation Results with Label Powerset transformation

Data set Emotions								
	C4.5	CO^2RBFN	KNN	MLP	Naive Bayes	PART	RBFN	SVM
Hamming Loss	0.277	0.243	0.235	0.234	0.233	0.293	**0.217**	0.281
Subset Accuracy	0.207	**0.301**	0.268	0.278	0.268	0.209	0.298	0.271
Example-Based Recall	0.541	**0.653**	0.626	0.630	0.630	0.526	0.6469	0.595
Example-Based Accuracy	0.438	0.522	0.514	0.518	0.512	0.424	**0.542**	0.473
Data set Scene								
	C4.5	CO^2RBFN	KNN	MLP	Naive Bayes	PART	RBFN	SVM
Hamming Loss	0.144	0.186	0.111	0.114	0.137	0.139	0.116	**0.095**
Subset Accuracy	0.547	0.427	0.629	0.641	0.537	0.563	0.621	**0.688**
Example-Based Recall	0.609	0.454	0.693	0.701	0.678	0.626	0.677	**0.720**
Example-Based Accuracy	0.589	0.454	0.674	0.684	0.615	0.605	0.662	**0.720**

sults similar to other methods. The worst results for CO^2RBFN are for the LP transformation of Scene. In any case, CO^2RBFN is the method with more best results in individual measures, along with SVM.

In summary, when transformations are applied to multi-label data sets in order to solve the associated classification problem, the behavior of classical algorithms can be irregular. The reasons can be found in the drawbacks of these transformations, mentioned in Section 2, or in intrinsic characteristics of the multi-label data set.

5 Conclusions

In many real classification data sets, instances can be associated to more than one class. These data sets are called multi-label data sets. Examples of related applications are: text categorization or semantic association of audio and video. We can distinguish two ways to solve a multi-label problem: Problem Transformation and Algorithm Adaptation. With the first approach the original data set is transformed in single-label data-sets in order to apply traditional classification methods. The other way involves adapting classical algorithms to manage multi-label data.

In this paper, a first approach to multi-label classification, CO^2RBFN, a cooperative-competitive evolutionary model for the design of RBFNs, is tested with multi-label data sets. The results of CO^2RBFN, and other data mining methods chosen for comparison, show some irregularities in their performance depending on the transformation, data set or measure used. This behavior may be due to the drawbacks described for transformation problem methods or to the intrinsic characteristics of the multi-label data sets.

As future line, we propose a deep analysis of the multi-label problem in order to carry out our developments, taking into account characteristics such as high dimensionality, correlations among labels or the interpretability of results obtained.

Acknowledgments: Supported by the Spanish Ministry of Science and Technology under the Project TIN2008-06681-C06-02, FEDER founds, and the Andalusian Research Plan TIC-3928.

References

1. D. Broomhead and D. Lowe. Multivariable functional interpolation and adaptive networks. *Complex Systems*, 2:321–355, 1988.
2. O. Buchtala, M. Klimek, and B. Sick. Evolutionary optimization of radial basis function classifiers for data mining applications. *IEEE Transactions on System, Man, and Cybernetics, B*, 35(5):928–947, 2005.
3. A.C.P.L.F Carvalho and A. A. Freitas. *Foundations of Computational Intelligence Vol. 5. Studies in Computational Intelligence 205*, chapter A tutorial on multi-label classification techniques, pages 177–195. Springer, 2009.
4. A. Clare and R. King. Knowledge discovery in multi-label phenotype data. *Proceedings of the 5th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD 2001)*, pages 42–53, 2001.
5. A. Elisseeff and J. Weston. A kernel method for multi-labelled classification. *Advances in Neural Information Processing Systems*, 14:–, 2002.
6. D. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley, Reading, MA, 1989.
7. M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I.H. Witten. The weka data mining software: An update. *SIGKDD Explorations*, 11(1), 2009.
8. C. Harpham, C.W. Dawson, and M.R. Brown. A review of genetic algorithms applied to training radial basis function networks. *Neural Computing and Applications*, 13:193–201, 2004.
9. E. Mandani and S. Assilian. An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, 7(1):1–13, 1975.
10. M.D. Pérez-Godoy, A.J. Rivera, M.J. del Jesus, and F.J. Berlanga. CO^2RBFN: An evolutionary cooperative-competitive RBFN design algorithm for classification problems. *Soft Computing*, 14(9):953–971, 2010.
11. G. Tsoumakas, I. Katakis, and I. Vlahavas. *Data Mining and Knowledge Discovery Handbook*, chapter Mining Multi-label Data, pages 667–668. Springer, 2nd edition, 2010.
12. B. Whitehead and T. Choate. Cooperative-competitive genetic evolution of radial basis function centers and widths for time series prediction. *IEEE Transactions on Neural Networks*, 7(4):869–880, 1996.
13. B. Widrow and M.A. Lehr. 30 years of adaptive neural networks: perceptron, madaline and backpropagation. *Proceedings of the IEEE*, 78(9):1415–1442, 1990.
14. M. L. Zhang. Ml-rbf : Rbf neural networks for multi-label learning. *Neural Processing Letters*, 29(2):61–74, 2009.
15. M. L. Zhang and Z. H. Zhou. Ml-knn: A lazy learning approach to multi-label learning. *Pattern Recognition*, 40:2038–2048, 2007.
16. Y. Zhang, S. Burer, and W. N. Street. Ensemble pruning via semi-definite programming. *Journal of Machine Learning Research*, 7:1315–1338, 2006.